AFWAL-TR-82-3098
Parts I,II,III,IV
Addendum 1

AD-A169 383

MAGNA (Materially and Geometrically Nonlinear Analysis)

R.A. Brockman

University of Dayton Research Institute
300 College Park Avenue
Dayton, Ohio 45469

May 1986

Final Report

DTIC FILE COPY

Approved for public release; distribution unlimited.

FLIGHT DYNAMICS LABORATORY
AIR FORCE AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6553

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Office of Public Affairs (ASD/PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

Richard A. Smith

RICHARD A. SMITH
Project Engineer

Robert E. McCarty

ROBERT E. MCCARTY
Group Leader
Subsystems Development Group

FOR THE COMMANDER

SOLOMON R. METRES
Chief
Vehicle Equipment Division

"If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify AFWAL/FIER, W-PAFB OH 45433 to help us maintain a current mailing list."

Copies of this report should not be returned unless return is required by security considerations, contractural obligations, or notice on a specific document.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE **AD-A169 383**

1a. REPORT SECURITY CLASSIFICATION: Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for public release; distribution unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S): AFWAL-TR-82-3098, Parts I-IV, Addendum 1
6a. NAME OF PERFORMING ORGANIZATION: University of Dayton Research Institute; 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION: AFWAL/FIER
6c. ADDRESS (City, State and ZIP Code): 300 College Park Avenue, Dayton, Ohio 45469	7b. ADDRESS (City, State and ZIP Code): Wright-Patterson AFB, Ohio 45433
8a. NAME OF FUNDING/SPONSORING ORGANIZATION; 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: F33615-82-K-3402
8c. ADDRESS (City, State and ZIP Code)	10. SOURCE OF FUNDING NOS.

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT NO.
62201F	2402	03	32

11. TITLE (Include Security Classification) MAGNA (Materially and Geometrically Nonlinear Analysis)

12. PERSONAL AUTHOR(S) R.A. Brockman

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Yr., Mo., Day)	15. PAGE COUNT
Final	FROM 9/82 TO 3/84	1986 May	195

16. SUPPLEMENTARY NOTATION The computer program contained in this technical report is theoretical and in no way reflects any Air Force owned software programs.

17. COSATI CODES		
FIELD	GROUP	SUB. GR.

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Nonlinear Structural Analysis; Numeriacl Analysis; Finite Element Analysis; MAGNA; Plasticity; Aircraft Windshield

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

The finite element program MAGNA (Materially And Geometrically Nonlinear Analysis), has been developed for nonlinear static and dynamic analysis of complex three-dimensional structures. This program is applicable to large structural response problems involving bars, membranes, beams, plates, shells, and three-dimensional solids, experiencing large displacements, finite strains, large rotations, and plastic deformation. This revision package reflects a number of additions and modifications to the MAGNA software. These include the migration of CDC program versions from NOS/BE to the NOS operating system, conversion of source code to FORTRAN-77 (ANSI X3.9-1978), elimination of file naming and numbering conflicts among different program options, usage of CRAY COS V1.11 which is compatible with both CRAY-1 and CRAY-X/MP computers, conversion to IBM computers under the OS/VS2 MVS operating system, modification of VAX/VMS and CRAY restart(over)

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT: UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS ☐	21. ABSTRACT SECURITY CLASSIFICATION
22a. NAME OF RESPONSIBLE INDIVIDUAL: R.A. Brockman	22b. TELEPHONE NUMBER (Include Area Code): 513-255-2516; 22c. OFFICE SYMBOL: AFWAL/FIER

DD FORM 1473, 83 APR EDITION OF 1 JAN 73 IS OBSOLETE. UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

Block 19

options, addition of a cyclic symmetry feature in natural frequency analysis with an option for static condensation.

Accession For	
NTIS GRA&I	X
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

QUALITY INSPECTED

AFWAL-TR-82-3098, Parts I-IV

SUMMARY OF ADDENDUM CHANGES

May 1986

The attached Revision "A" change package reflects the following additions and modifications to the MAGNA software:

1. Migration of CDC program versions from NOS/BE to the NOS operating system
2. Conversion of source code to FORTRAN-77 (ANSI X3.9-1978)
3. Elimination of file naming and numbering conflicts among different program options
4. Usage of CRAY COS V1.11, compatible with both CRAY-1 and CRAY-X/MP computers
5. Conversion to IBM computers, under OS/VS2 MVS operating system
6. Modification of VAX/VMS and CRAY restart options
7. Addition of cyclic symmetry feature in natural frequency analysis, with option for static condensation

The primary impact of these software upgrades is on the finite element analysis procedure, which is documented in Part I of the subject report. For Part I, change pages are provided in the Revision package, which should be inserted as indicated in the original report.

Pre- and post-processing operations, documented in Parts II and III of the report, have been affected only slightly. The primary difference which affects routine usage is the method of accessing the program files under NOS on CDC computer systems. Examples which illustrate revised access procedures for preprocessing and postprocessing utilities are included as Addenda to Parts II and III of the report.

AFWAL-TR-82-3098, Part I

INSTRUCTIONS FOR ADDENDUM CHANGES

May 1986

Remove Pages	Insert Pages	Description of Changes
7.0.1	7.0.1	Revised chapter introduction
7.1.1 - 7.1.18	7.1.1 - 7.1.18	Changes for CDC NOS systems; file changes for FORTRAN-77
7.2.1 - 7.2.14	7.2.1 - 7.2.12	Changes for COS 1.11 and UIS; file changes for FORTRAN-77
---	7.4.1 - 7.4.63	Program usage on IBM systems
8.2.2 - 8.2.4	8.2.2 - 8.2.4	Cyclic symmetry option input
8.2.6 - 8.2.7	8.2.6 - 8.2.8	Notes on cyclic symmetry data
8.3.4	8.3.4-1 - 8.3.4-3	Cyclic symmetry solution data
---	8.3.6a - 8.3.6c	Notes on cyclic symmetry data
8.3.7	8.3.7	Restart data changes for VAX and CRAY under FORTRAN-77
8.3.9 - 8.3.11	8.3.9 - 8.3.11	(Section removed)
---	8.7.4 - 8.7.15	Cyclic symmetry data
8.8.2	8.8.2	Notes on constraint data for cyclic symmetry solution
---	R.7.1	References for IBM control language procedures

AFWAL-TR-82-3098, Part II

INSTRUCTIONS FOR ADDENDUM 1 CHANGES

May 1986

The preprocessing examples included in this addendum are revised versions of the example problem listings of subsections 7.1 and 7.3. The revisions reflect changes in preprocessing procedures under the NOS operating system at the ASD Computer Center, Wright-Patterson Air Force Base.

Pre- and post-processing utilities related to the MAGNA finite element program are now accessible through a single CCL procedure file, UTILS. This change, and minor modifications to file access procedures necessitated by the migration to NOS, are the primary modifications to the software user interface. The preprocessing utilities remain functionally intact.

Remove Pages	Insert Changes	Description of Changes
---	Part II Addendum (7.3.1-7.3.31) (7.56.1-7.56.31)	Changes for NOS Operating System on CDC Computers

AFWAL-TR-82-3098, Part III

Postprocessor Manual

ADDENDUM

INSTRUCTIONS FOR ADDENDUM CHANGES

May 1986

Pre- and post-processing utilities related to the MAGNA finite element program are now accessible through a single CCL procedure file, UTILS. This change in operating procedures coincides with the migration to the NOS operating system on the CDC systems at the ASD Computer Center, Wright-Patterson Air Force Base. Access to postprocessing functions through UTILS, as well as minor changes in file access procedures necessitated by the migration to NOS, are the primary modifications to the software user interface. The preprocessing utilities remain functionally intact.

The examples which follow illustrate typical procedures for accessing the UTILS procedure interactively and in batch mode (for the CPLOT DISSPLA output option). During interactive use, initial access to the UTILS procedure and selection of utility functions from the menus represent the only major changes to postprocessing operations under the NOS operating system.

Remove Pages	Insert Changes	Description of Changes
---	Part III Addendum	Changes for NOS Operating System on CDC Computers
	93	
Change page nrs. 93-94	to 94-95	

CHAPTER 7

PROGRAM OPERATION

PART II

The MAGNA finite element program is currently operational on the following computer systems:

- Control Data CDC 6000 and CYBER series under NOS;
- Cray Research CRAY-1 series and CRAY-X/MP under COS;
- Digital Equipment VAX-11 series under VAX/VMS; and
- IBM 3081 under OS/VS2.

Execution of the program on each of these computer systems is outlined in detail in this Chapter. Information to be used in estimating execution times for the various analysis options are also presented where possible for all machine versions.

7.1 CDC PROGRAM VERSION

The CDC version of MAGNA, excluding pre- and postprocessors, is accessed through two control procedures, MAGNJCL and STRAVG. These two procedures control the execution of MAGNA and STRAVG, respectively. Modification of program storage capacity and insertion of user-written subroutines are performed automatically based on simple data and keywords supplied as part of the input job stream.

7.1.1 Job Control Language

The CDC computer version of MAGNA is typically executed using a job stream of the form shown below. Parameters which are replaced by user-supplied names or data are shown in lower case letters; upper case text is entered exactly as shown.

```
 1.  /JOB
 2.  jobname.
 3.  /USER
 4.  CHARGE,*.
 5.  GET,TAPE5=filename1.
 6.  GET,USRSUB=filename2.
 7.  DEFINE,MPOST=filename3.
 8.  GET,P=MAGNJCL/UN=D820139.
 9.  SETTL,timelimit.
10.  BEGIN,XMAGNA,P,MAIN,USRSUB.
11.  DEFINE,APOST=filename4.
12.  GET,STRVJCL/UN=D820139.
13.  BEGIN,STRAVG,STRVJCL.
14.  /EOR
15.  (STORAGE ALLOCATION card)
16.  /EOR
17.  /EOF
```

An input file such as the one above is submitted for processing using the interactive command:

/SUBMIT,lfn,BC

where "lfn" is the name of the local file on which the job stream is stored.

The sample job stream above exercises optional features which may not be used at all times. These include user written subroutines, modification of storage capacity, execution of the STRAVG postprocessor, and saving the results files MPOST and APOST as permanent disk files. Some other options, notably restart, are not shown above; analysis restarts are discussed separately at the end of this section.

For routine analyses in which the optional features mentioned above are not needed, the job control file has a much simpler form:

```
 1.  /JOB
 2.  jobname.
 3.  /USER
 4.  CHARGE,*.
 5.  GET,TAPE5=filename.
 8.  GET,P=MAGNJCL/UN=D820139.
 9.  SETTL,timelimit.
10.  BEGIN,XMAGNA,P.
14.  /EOR
17.  /EOF
```

The function of each statement in the MAGNA control deck is explained in detail below.

CARD 1: /JOB. The job card identifies the input file as a "SUBMIT" file. Of the two batch job formats which are

available under NOS (SUBMIT and ROUTE), the SUBMIT format is preferred for reasons of password security.

CARD 2: jobname. This line defines the user job name, which consists of a character string up to 7 characters in length.

CARD 3: /USER Directive. The /USER directive supplies the account number and password for the job to be run. When the file is SUBMITted, this directive is automatically replaced with the current account number and password. It can be replaced by an explicit USER statement, which has the form USER,account,password.

CARD 4: CHARGE Card. The statement "CHARGE,*." causes the job to be charged to the default account under which it was submitted. To charge job costs to a different account, use the statement "CHARGE,chargenumber,projectnumber.".

CARD 5: GET,TAPE5. This GET statement retrieves a copy of the input data file (see Chapter 8 of this manual) as a local file. Data can also be copied directly from the input file using "COPYCR,INPUT,TAPE5.".

CARD 6: GET,USRSUB. This GET statement retrieves a file containing the FORTRAN source code for optional user written subroutines. Note that user subroutines should be coded in FORTRAN 77 (CDC FORTRAN 5) to avoid complications from intermixed binary files.

CARD 7: DEFINE,MPOST. This control statement is used whenever the MPOST postprocessor file is to be saved on disk following execution of MAGNA. DEFINE identifies the MPOST file as a direct access file, which will be saved automatically after the job is complete. The name MPOST is a local file name used only during the run; "filename3" will be the permanent

file name assigned to the postprocessor file. An MPOST file can be saved directly on magnetic tape by replacing the DEFINE command by a LABEL command; for example:

LABEL,MPOST,VSN=xxxxxx,D=GE,L=label,PO=AW,W.

Note that the RESOURC command may be necessary (e.g., RESOURC, GE=2) when requesting multiple tapes during a single job. Writing the MPOST file directly to tape is not suggested if the STRAVG utility is to be used in the same run. Instead, MPOST should be saved on disk and copied to tape following the BEGIN statement (CARD 10).

CARD 8: GET,P=MAGNJCL. This GET command attaches the MAGNA control procedure as a local file. User numbers (UN=) under which the procedure is stored are installation-dependent.

CARD 9: SETTL. The SETTL command sets a CPU time limit (in seconds) for the job step in which MAGNA is executed. The time limit is expressed in seconds; for instance, SETTL,60. limits the execution step to one minute.

CARD 10: BEGIN,XMAGNA. The BEGIN statement initiates execution of MAGNA. The keyword MAIN causes MAGNA to read the STORAGE ALLOCATION card (CARD 15), and to modify the storage capacity of the program. Including "USRSUB" causes MAGNA to look for a local file of this name, compile it under FTN5, and relink using the user subroutines contained in the file. Possible forms of the BEGIN command are:

```
BEGIN,XMAGNA,P.
BEGIN,XMAGNA,P,MAIN.
BEGIN,XMAGNA,P,,USRSUB.
BEGIN,XMAGNA,P,MAIN,USRSUB.
```

CARD 11: DEFINE,APOST. This command is needed if the stress smoothing utility STRAVG is to be executed. Note that an MPOST postprocessing file must be created in order to use STRAVG. The DEFINE statement identifies the file APOST as a direct access file, which will be saved automatically when the job is complete. This DEFINE command can be replaced by a LABEL command (see CARD 7, DEFINE MPOST) to store the APOST file directly on magnetic tape.

CARD 12: GET,STRVJCL. This command accesses the control procedure for STRAVG, a stress-smoothing utility (see Section 5.7). User names (UN=) under which STRAVG is installed are installation-dependent.

CARD 13: BEGIN,STRAVG. This BEGIN statement initiates execution of the STRAVG postprocessor. In the CDC version of MAGNA, three versions of STRAVG with differing storage capacity are available. The appropriate NOS control statements for each are:

```
BEGIN,STRAVG,STRVJCL.    (small capacity)
BEGIN,STRAVGL,STRVGJCL. (medium capacity)
BEGIN,STRAVGM,STRVJCL.   (large capacity)
```

All versions of STRAVG are capable of processing large finite element models; however, the larger-capacity versions are much more efficient for medium to large sized problems.

CARD 14: /EOR. This end-of-record marks the end of the job control statements for a MAGNA run.

CARD 15: STORAGE ALLOCATION CARD. The STORAGE ALLOCATION card is required whenever the program storage capacity is to be modified. The keyword "MAIN" in the BEGIN, XMAGNA statement causes this card to be read. The format of the card is (lx,6I5); that is, six integers of five digits each,

beginning in column 2 of the line. Contents of the data fields and their default values are described in Section 7.1.2.

CARD 16: /EOR. This end-of-record is required whenever the STORAGE ALLOCATION CARD is included in the MAGNA input stream.

CARD 17: /EOF. The end-of-file marks the end of the input stream, and is always required.

Some files, such as problem input data and user written subroutines, can be copied directly from the input stream if this is more convenient. The sample job below copies both files from input, and shows an example of the STORAGE ALLOCATION card used to modify program capacity. It also illustrates typical procedures for saving the MPOST and APOST postprocessor files on magnetic tape following execution.

```
/JOB
RABMAG.
/USER
CHARGE,*.
COMMENT.*******************
COMMENT.***** NO DECK *****
COMMENT.*******************
COPYCR,INPUT,TAPE5.
COPYCR,INPUT,MYSUB.
REWIND,MYSUB.
GET,P=MAGNJCL/UN=D820139.
SETTL,1000.
BEGIN,XMAGNA,P,MAIN,MYSUB.
GET,STRVJCL/UN=D820139.
BEGIN,STRAVGL,STRVJCL.
LABEL,PPTAPE,VSN=L12345,D=GE,L=EXAMPLE,PO=AW,W.
```

```
REWIND,MPOST,APOST.
COPYBF,MPOST,PPTAPE.
COPYBF,APOST,PPTAPE.
/EOR
     :
     :
  (input data)
     :
     :
/EOR
     :
     :
  (user subroutines)
     :
     :
/EOR
40000 3200
/EOR
/EOF
```

Many nonlinear or transient dynamic solutions are best performed in more than one submission of the program, in order to

- monitor progress of the solution,
- reduce computer resources for individual runs,
- safeguard against system failure, and
- modify the data or solution method.

Optionally, MAGNA will create restart checkpoints at the conclusion of specified increments during an analysis; the problem can be restarted from any of these points in a later job. The input data needed to request the creation of restart files is described in Section 8.3; Section 5.8 discusses the restart capabilities of MAGNA, including types of data which

may be changed in a restart run.

The job control language needed to perform a restart consists of supplying the existing restart file (if it exists) to the program as a local file named NOREST, and saving newly created restart files, which are written to the file NRSTAP. More than one restart file can be written during one analysis job; each such file is a single system logical record on the restart tape. Both old and new restart files can be stored on the same tape; however, precautions should be taken not to read data from an existing tape with a write-enable ring in place. If a particular tape is to be read and written in a single job, it is good practice to remount the tape between the read and write operations, removing or inserting the ring as necessary. Several examples are given below to demonstrate the necessary job control statements for restart runs.

Case 1: New Analysis; Creation of Restart Tape

In this example a new restart tape is written directly to magnetic tape. Since this is a new analysis, no old restart tape is needed.

```
LABEL,NRSTAP,VSN=L12345,D=GE,L=RESTART1,PO=AW,W.
  :    :    :
  :    :    :
BEGIN,XMAGNA,P.
```

Case 2: Analysis Restart; No New Restart Tape

This example shows how an existing restart tape must be accessed in order to resume a previous analysis. The new run is to restart using the ninth restart file on the tape; therefore eight restart files (system logical records) must be skipped before the tape is read. In the example, the

restart data is copied from the tape to a local file, so the tape drive can be released to the system before MAGNA executes.

```
LABEL,RSTAPE,VSN=L13579,D=GE,L=REST0121,PO=AR,R.
SKIPR,RSTAPE,8.
COPYBR,RSTAPE,NOREST.
RETURN,RSTAPE.
REWIND,NOREST.
   :     :     :
   :     :     :
BEGIN,XMAGNA,P.
```

Case 3: Both Old and New Restart Tapes Used

This example illustrates the use of both old and new restart tapes in the same run. Both files will reside on the same reel of tape. The old restart file to be read is the twelfth restart file on the tape, and the new restart file is to be stored as the twentieth file on the same tape.

```
LABEL,MTAPE,VSN=X24680,D=GE,L=RSIMPACT,PO=AR,R.
SKIPR,MTAPE,11.
COPYBR,MTAPE,NOREST.
RETURN,MTAPE.
LABEL,NRSTAP,VSN=X24680,D=GE,L=RSIMPACT,PO=AW,R.
SKIPR,NRSTAP,19.
   :     :     :
   :     :     :
BEGIN,XMAGNA,P.
```

The second type of restart function performed by MAGNA is the eigenvalue solution with prestress effects (see Sections 4.5, 5.9, and 8.3). With this option, a nonlinear solution is first performed to determine the equilibrium state.

Stiffness coefficients from the nonlinear analysis, which include the effect of static stresses, large deflections, and material yielding, are then incorporated in a subsequent natural frequency analysis. Physically, the natural frequency solution represents small-amplitude harmonic vibration which is superimposed on the deformed configuration.

The two sample job decks listed below demonstrate the use of the eigenvalue-with-prestress analysis option. In the first run, which is nonlinear, two files are saved: file STIFF contains the element stiffness coefficients, and MPOST is the results file containing the deformed geometry and static stresses. The second run, a natural frequency analysis, must access these files to perform the free vibration solution. The use of file MPOST from the first analysis is optional; it will cause the natural frequency run to create an MPOST file in which the statically deformed geometry is stored as the "undeformed", or reference, geometry, and the displacements are those determined by the vibration mode shapes.

Run No. 1: Nonlinear Analysis of Prestress State

```
/JOB
RAB1.
/USER
CHARGE,*.
GET,TAPE5=NLDATA.
DEFINE,STIFF=NLSTIF.
DEFINE,MPOST=NLMPST.
GET,P=MAGNJCL/UN=D820139.
SETTL,800.
BEGIN,XMAGNA,P,MAIN.
/EOR
40000 3500
/EOR
/EOF
```

Run No. 2: Frequency Analysis with Prestress

```
/JOB
RAB2.
/USER
CHARGE,*.
GET,TAPE5=FRQDAT.
ATTACH,STIFF=NLSTIF.
ATTACH,MPOLD=NLMPST.
DEFINE,MPOST=NFMPST.
GET,P=MAGNJCL/UN=D820139.
SETTL,600.
BEGIN,XMAGNA,P,MAIN.
/EOR
60000 3500
/EOR
/EOF
```

7.1.2 Modification of Storage Capacity

MAGNA allocates array storage dynamically for all matrices and internal tables whose size is problem-dependent. Although analyses of rather large size can be accomplished with only a small amount of array space, efficiency is improved dramatically by allocating additional storage for larger problems. Modification of the program's storage capacity is quite simple, since only one additional data card is needed in the input deck (see Section 7.1.1, Card 15, STORAGE ALLOCATION card).

Six parameters control the storage capacity of MAGNA; these parameters define the lengths of the five labeled COMMON blocks described below.

1. /BLANK/ - contains most of the large arrays and internal tables, including assembled stiffness, mass or effective stiffness matrix partitions.

2. /IDENT/ - contains bookkeeping information about active nonzero terms of the stiffness matrix.

3. /BLOX/ - contains tables which control the out-of-core storage of system matrices.

4. /BLEQ/ - contains tables which control the out-of-core storage of system matrices.

5. /RAF21/ - contains record keys for direct access file I/O.

6. /USERC/ - contains working space available for use by user-written subroutines.

This ordering corresponds to the six integer data fields of the STORAGE ALLOCATION card. The minimum and default lengths of each block are summarized in Table 7.1.1.

COMMON blocks /BLANK/ and /IDENT/ determine the in-core storage capacity of the program. The length of /BLANK/, which contains partitions of the system matrices, is dictated by problem size and the density of the stiffness matrix. For one- and two-dimensional models, the default /BLANK/ size is normally sufficient. For larger models, particularly those using three-dimensional elements, I/O efficiency is improved substantially by extending the length of COMMON /BLANK/.

The length of COMMON /IDENT/ must be greater than the total number of unknowns in the model. An upper bound on

TABLE 7.1.1

DEFAULT AND MINIMUM COMMON BLOCK LENGTHS
(CDC Program Version)

BLOCK	DEFAULT LENGTH	MINIMUM LENGTH
/BLANK/	20000	12000
/IDENT/	2500	100
/BLOX/	150	150
/BLEQ/	150	150
/INDXK/	170	170

the space required in /IDENT/ is (number of degrees of freedom per node)x(number of nodes).

The lengths of COMMON blocks /BLOX/, /BLEQ/ and /RAF21/ determine the out-of-core storage accessible by MAGNA. These blocks should remain at the default values for all but the largest three-dimensional problems. When problem size capacity must be increased, extending the main memory space (by modifying /BLANK/ and /IDENT/) is always preferred.

The length of COMMON /USERC/ is dictated by the requirements of user-written subroutines, since this block is not used internally by MAGNA. Section 9.3 contains an example in which block /USERC/ is used to retain information for use by a user subroutine.

7.1.3 Reserved File Names

Since MAGNA executes under the control of command files which automatically attach and manipulate files which are required for an analysis, certain file names used in the control procedure are reserved and may not be used elsewhere. The following file names should not be in use when the BEGIN command is issued:

ABS	NEWPL
COMPILE	OLDPL
ERRORS	SEGLOD
MAGNA	TEMP
MAIN	UPDGEN
MODS	UPDIN
NEWB	USUB

TABLE 7.1.2

OCTAL-DECIMAL CONVERSIONS

Decimal	Octal	Octal	Decimal
1000	1750	1000	512
5000	11610	10000	4096
10000	23420	60000	24576
20000	47040	100000	32768
30000	72460	120000	40960
40000	116100	140000	49152
50000	141520	160000	57344
60000	165140	200000	65536
70000	210560	220000	73728
80000	234200	240000	81920
90000	257620	260000	90112
100000	303240	300000	98304

7.1.4 Typical Execution Times on CDC Computers

Data are presented in this section to aid in the estimation of computer run times on CDC machines using MAGNA. The times, formulas and data given are based upon observed execution times on the CDC 6600 computer. For the CYBER 74 model, run times are nearly identical. On the CYBER 175/750, CPU times are generally less than half the CDC 6600 time, and I/O times may be slightly less.

In nonlinear analysis, computing times are often dominated by the number of elements rather than the time for solving the matrix equations. This is always true for three-dimensional elements, where the nonlinear element calculations are extremely complex. Computing time factors for each of the MAGNA elements appear in Table 7.1.3; for most nonlinear analyses, the CPU time can be estimated conservatively using the formula:

$$\begin{aligned}\text{CPU time} = {} & (\text{CPU Time Factor}) \times (\text{Number of Elements}) \\ & \times (\text{Number of Integration Points / Element}) \\ & \times (\text{Number of Increments})\end{aligned}$$

where the CPU time factor is read from the Table. An overhead of approximately 15-20% should be added to this amount for the solution of equations and other calculations. The IO/CPU ratio from the Table can be used next to estimate the IO time needed for the job. For nonlinear analyses using equilibrium iteration each iteration cycle should be counted as an "increment" in estimating the solution time. The resulting estimate will be quite conservative, since iteration cycles take less time than a simple increment without iteration.

Computation times for nonlinear dynamic analysis are only slightly higher than for nonlinear static analysis, and the above estimating procedure can be used with confidence.

TABLE 7.1.3

COMPUTING TIME FACTORS FOR INDIVIDUAL ELEMENT TYPES (CDC 6600)

Element Type	CPU Sec/Integration Point/Increment		IO/CPU Time Ratio	
	Linear	Nonlinear	Linear	Nonlinear
1	0.04	0.20	2.0-6.0	1.0-1.4
2	0.01	0.04	2.0-6.0	2.0-4.0
3	0.04	0.01	4.0-7.0	4.0-10.0
4	0.01	0.01	4.0-10.0	4.0-10.0
5	0.02	0.08	1.0-6.0	2.0-4.0
6	0.03	0.11	2.0-8.0	1.5-3.5
7	0.03	0.11	2.0-8.0	1.5-3.5
8	0.02	0.07	2.0-10.0	2.0-5.0
9	.01	.01	3.0-7.0	4.0-8.0
10	.01	.01	3.0-7.0	4.0-8.0

Note that in elastic-plastic analysis, the amount of computing per element may vary drastically, and estimation of execution times is necessarily less accurate. It may be advisable to inflate the above CPU estimates by 25-30% if strong material nonlinearity is expected; I/O times are not affected by the occurence of material nonlinearities.

For linear analysis, the estimation of computer resources is much more difficult, since solution times are dominated by the assembly and solution of the linear equations. Values of the CPU time factor given in Table 7.1.3 refer to element calculations only, and do not give a reliable estimate of the total time required for a linear static analysis. The IO/CPU time ratios which appear in the Table are reasonably accurate. The higher values of IO/CPU times apply for linear dynamic analysis, where CPU times are typically quite modest.

7.2 CRAY PROGRAM VERSION

The CRAY-1 and CRAY-X/MP versions of MAGNA offer the largest problem size capacity and highest speed of all the machine versions available. Procedures for running MAGNA on the CRAY machines are described in this section.

Most CRAY installations do not provide for interactive operation. Instead, job streams and data files are prepared on a front-end machine and submitted to the CRAY computer through remote job entry facilities. For this reason, the operation of MAGNA on CRAY systems is highly installation-dependent. The information in this section is sufficient to prepare the proper CRAY job control language in most cases; for further details on job submittal, magnetic tape access, or commands for file operation (on system_ with shared disk resources), it may be necessary to consult systems support personnel at the particular installation in question.

The CRAY job control language used in examples below is compatible with COS Version 1.11, on both the CRAY-1 and CRAY-X/MP computers.

7.2.1 Job Control Language

The CRAY computer version of MAGNA is typically executed using a job control stream of the form shown below.

```
1.  JOB,JN=jobname,Tnnn.
2.  ACCOUNT,AC=acctno,US=userno,UPW=password.
3.  ASSIGN,A=FT10,DN=RF10,BS=32,RDM.
4.  ASSIGN,A=FT21,DN=RF21,BS=4,RDM.
5.  ASSIGN,A=FT12,DN=STIFF.
6.  ASSIGN,A=FT55,DN=NPREST.
7.  ASSIGN,A=FT92,DN=MPOLD.
8.  ASSIGN,A=FT95,DN=NSUBOL.
```

9. ASSIGN,A=FT96,DN=NTSUB.
10. ASSIGN,A=FT97,DN=NOREST.
11. ASSIGN,A=FT98,DN=NRSTAP.
12. ASSIGN,A=FT99,DN=MPOST.
13. ACCESS,DN=NPREST,PDN=filename.
14. ACCESS,DN=MPOLD,PDN=filename.
15. ACCESS,DN=NSUBOL,PDN=filename.
16. ACCESS,DN=NOREST,PDN=filename.
17. ACCESS,DN=FT05,PDN=filename.
18. ACCESS,DN=MAGNA,ID=id,OWN=owner.
19. MAGNA.
20. SAVE,DN=STIFF,PDN=filename.
21. SAVE,DN=NTSUB,PDN=filename.
22. SAVE,DN=NRSTAP,PDN=filename.
23. SAVE,DN=MPOST,PDN=filename.
24. EOF

The function of each entry in the MAGNA job control file is explained in more detail below.

LINE 1: JOB CARD. The JOB card identifies the start of the job stream, provides the job name ("jobname"), and requests resources for the run. Note that the CPU time limit is an octal number; for example, T100 sets a time limit for the job at 64 seconds.

LINE 2: ACCOUNT CARD. The ACCOUNT card defines the user number under which the job is to be processed.

LINES 3-4: ASSIGN,...,RDM. These two statements are always required. They define the two direct access files used by MAGNA, and declare buffer sizes for these files.

LINES 5-12: ASSIGN,... This series of ASSIGN statements define sequential files which may be used in the MAGNA run. Each of these files may be opened with a STATUS

other than "SCRATCH", depending upon specific options selected in the data. They include restart, postprocessing, and other files which may be saved as permanent datasets in connection with a MAGNA job. It is suggested that all eight ASSIGN statements be used at all times, to eliminate the possibility of errors.

LINE 13: ACCESS,DN=NPREST. This control statement is necessary only if the eigenvalues-with-prestress option (see Section 5.9) is being used. The file "filename" is the stiffness file (STIFF) saved at the end of a prior nonlinear analysis.

LINE 14: ACCESS,DN=MPOLD. This statement is also used in the eigenvalue step of the frequencies with prestress option. File "filename" is the MPOST file from the preceding nonlinear analysis. This file is the "prestressed geometry file" defined for plotting purposes in Section 8.3 (INCPRE>0).

LINE 15: ACCESS,DN=NSUBOL. This control statement is used to retrieve a previously-created substructure file (option not currently active).

LINE 16: ACCESS,DN=NOREST. This entry is used to attach the "old" restart file when restarting a nonlinear analysis. If no old restart file is requested in the data, the statement is not needed.

LINE 17: ACCESS,DN=FT05. This statement accesses the problem data (as described in Chapter 8) for the current MAGNA run. The file containing the data is logical unit 5; note that this is NOT the same as the default input dataset, $IN. The ACCESS statement may be replaced by a command to COPY the input data from the job file, but file FT05 must be rewound following the COPY command.

LINE 18: ACCESS,DN=MAGNA. The executable version of MAGNA is accessed through this command. The parameters "id" and "owner" are installation-dependent.

LINE 19: MAGNA. This statement initiates MAGNA execution.

LINE 20: SAVE,DN=STIFF. This SAVE command applies only for nonlinear analyses, when the element stiffness file is to be saved for use in a subsequent analysis for frequencies with prestress. Since the element stiffness file may be quite large for three-dimensional models, it may be advisable to save this file on magnetic tape rather than disk.

LINE 21: SAVE,DN=NTSUB. This statement saves a substructure data file for later use (option not currently active).

LINE 22: SAVE,DN=NRSTAP. In nonlinear analysis or linear dynamic analysis, this command must be used to save the analysis restart file (see Section 5.8). The restart file may be large for three-dimensional models, and it may be advisable to save the file on magnetic tape.

LINE 23: SAVE,DN=MPOST. This command is used to save the postprocessing file "MPOST", which contains analysis results in a form suitable for plotting.

The control language listed above is adequate for most small and medium-sized problems. For larger models, and in nonlinear analysis, system defaults for maximum file size may be exceeded on the direct access file RF10. The maximum file size may be increased using the LM parameter of the ASSIGN statement; for example,

```
ASSIGN,A=FT10,DN=RF10,BS=32,RDM,LM=40000.
```

The value of LM is the maximum size of the file, expressed as a decimal number of 512-word blocks. The default and maximum values of LM are installation-defined.

7.2.2 Job Submittal under VAX/VMS Station Control

With the VAX/VMS station configuration, submitting MAGNA jobs to the CRAY system is quite simple, and the VAX/VMS station documentation (Cray Research Report No. SR-0020) is useful as a guide. Job control language as described above is prepared on the VAX computer, and the job is submitted to the CRAY using CSUBMIT:

```
$ CSUBMIT filename
```

Printed output from MAGNA normally appears on the VAX system as the file "jobname.CPR", where "jobname" is defined on the JOB statement.

Postprocessing files can be transferred from the CRAY to the VAX station using the DISPOSE statement under COS. For example, to save the MPOST file as a VAX/VMS file, the SAVE statement (line 23 above) might be replaced by:

```
DISPOSE,DN=MPOST,DC=ST.
```

The resulting VAX/VMS file would be named MPOST.CST;1. No reformatting (other than the station defaults) are required, since the postprocessor file is formatted and sequential. Saved files other than the MPOST file can remain in the CRAY binary blocked format, and need not be staged to the VAX/VMS station.

7.2.3 Job Submittal Under CDC Front-End Configuration

No standard front-end configuration based on the use of CDC equipment, so that job submittal procedures are entirely installation-dependent. The procedures described below are based upon the particular system configuration in use at United Information Systems, Inc. (a division of Control Data Corporation).

Once the MAGNA job control stream has been created on the CDC system, submittal of the job to the CRAY requires the use of a remote job entry command of one of the following forms:

RJE,F=filename,D=CRAY,CI=TTY. (APEX service)

or

CJE,F=filename,D=CRAY,CI=TTY. (SUPRA service)

Here "filename" is the name of the (local) file containing the CRAY job control stream. By default, the printed output will be spooled to the central site printer; to retain printed output for editing, or for printing at a local RJE terminal, the job stream must contain appropriate statements to SAVE the output file $OUT.

7.2.4 Local Modifications to COS at U.I.S.

Due to the common disk storage arrangement used for the CDC and CRAY systems at United Information Services, the standard COS control statements for file control (and several others) have modified formats. The COS control statements affected are: JOB, ACCOUNT, ACCESS, and SAVE. In addition, extra commands are necessary to save printed output on disk, and to save the dayfile from a CRAY job. The general form of the CRAY job stream on the UIS systems is shown below.

```
* 1.   JOB,Tnnn.
* 2.   ACCOUNT,userno,password.
  3.   ASSIGN,A=FT10,DN=RF10,BS=32,RDM.
  4.   ASSIGN,A=FT21,DN=RF21,BS=4,RDM.
  5.   ASSIGN,A=FT12,DN=STIFF.
  6.   ASSIGN,A=FT55,DN=NPREST.
  7.   ASSIGN,A=FT92,DN=MPOLD.
  8.   ASSIGN,A=FT95,DN=NSUBOL.
  9.   ASSIGN,A=FT96,DN=NTSUB.
 10.   ASSIGN,A=FT97,DN=NOREST.
 11.   ASSIGN,A=FT98,DN=NRSTAP.
 12.   ASSIGN,A=FT99,DN=MPOST.
*13.   GET,NPREST=filename.
*14.   GET,MPOLD=filename/CI=TTY.
*15.   GET,NSUBOL=filename.
*16.   GET,NOREST=filename.
*17.   GET,FT05=filename/CI=TTY.
*18.   GET,MAGNA/CRYLBRY.
 19.   MAGNA.
*20.   PUT,STIFF/D.
*21.   PUT,NTSUB/D.
*22.   PUT,NRSTAP/D.
*23.   PUT,MPOST/CO=TTY/D.
*23.1  DFD,MAGDAY,R.
*23.2  PUT,$OUT=MAGOUT/CO=TTY.
 24.   EOF
```

Statement numbering corresponds to that in Section 7.2.1, and modified statements are marked by an asterisk. The additional control statements (23.1, 23.2) save the job dayfile (MAGDAY) and the printed output (MAGOUT) on disk. Either or both of these lines may be omitted if these files are to be routed directly to the default printer.

The GET and PUT commands used above deserve some explanation. The command

GET,lfn=pfn.

attaches permanent file "pfn" to the job as the local dataset "lfn". Similarly,

PUT,lfn=pfn.

saves a local file named "lfn" on disk as file "pfn". The form "PUT,..../D." must be used for larger files, which are stored as direct access files; the size of indirect files (/D omitted) is limited, and omission of "/D" when PUTting a large file will result in a fatal error, terminating the job. Qualifiers "CI=TTY" or "CO=TTY" are necessary when the file in question is stored in the CDC file format (6- or 12-bit characters, 60 bits per word). When GETting a file created on the CDC computer, "/CI=TTY" causes the file to be translated to CRAY ASCII format (8-bit characters, 64 bits per word). With PUT, "/CO=TTY" causes the CRAY ASCII file to be reformatted in a form which can be printed (or read by postprocessors) on the CDC machine.

Magnetic tape files may also be staged on the CRAY system at UIS, using the control commands TAPEIN (copy a file from tape to a local file) and TAPEOUT (copy a local file to tape). The format of these commands has changed frequently with operating system upgrades. Consultation with the local UIS programmer/analyst is suggested to determine current tape access procedures under COS.

7.2.5 User-Written Subroutines

Use of user-written subroutines (see Chapter 9) in the CRAY version of MAGNA requires re-linking the program, with the user subroutines inserted. A MAGNA object library is maintained for this purpose. In the following, it is assumed that

FORTRAN source code for the user subroutines is stored on disk as file "USUBS".

For the standard CRAY operating system, the two control statements

```
18.  ACCESS,DN=MAGNA,ID=id,OWN=owner.
19.  MAGNA.
```

must be replaced by

```
17.1 ACCESS,DN=USUBS,PDN=USUBS.
17.2 CFT,I=USUBS,E=1.
17.3 REWIND,DN=$BLD.
18.  ACCESS,DN=$OBL,PDN=MAGOBJ,ID=id,OWN=owner.
18.1 BUILD,I=0,NODIR.
19.  LDR,DN=$NBL.
```

The CFT statement performs the FORTRAN compilation, resulting in a relocatable object file $BLD. The BUILD step updates the object library $OBL to include the user routines on file $BLD. Finally, LDR is used to link and execute the modified program.

For the modified COS installed at UIS, the control statements

```
18.  GET,MAGNA/CRYLBRY.
19.  MAGNA.
```

must be replaced by

```
17.1 GET,USUBS/CI=TTY.
17.2 CFT,I=USUBS,E=1.
17.3 REWIND,$BLD.
18.  GET,$OBL=MAGOBJ/CRYLBRY.
18.1 BUILD,I=0,NODIR.
19.  LDR,DN=$NBL.
```

The above example assumes that the user subroutines have been created on the CDC machine, and are stored in CDC display code. The qualifier CI=TTY causes automatic conversion of the file to CRAY ASCII format as the access takes place.

At most CRAY installations, the standard CFT compiler is CFT Version 1.10 or later. Versions of CFT later than 1.09 are FORTRAN 77 compilers, which requires that user-supplied subroutines must be written in FORTRAN 77. Some FORTRAN 66 constructs are supported under ANSI standard FORTRAN 77.

7.2.6 Modification of Storage Capacity

There is no user-controllable procedure for modifying the storage capacity of the CRAY-1 version of MAGNA. Each installation is configured with a near-maximum amount of working storage, determined by the amount of real memory on the system in question. If a larger-capacity CRAY version of MAGNA is required, arrangements must be made through the installation representative or the developers.

7.2.7 Execution Times on the CRAY-1 and CRAY-X/MP

Data collected from observed solution times on the CRAY-1 computer are summarized briefly in this section to aid in the estimation of computer run times for MAGNA. On the X/MP computer, the CRAY-1 execution times can be reduced by a factor of about 1.3.

In nonlinear analysis, computing time is typically dominated by the number of elements rather than by the time to solve equations. This observation is particularly true in three dimensional problems, due to the extreme complexity of the element calculations. Computing time factors on the CRAY-1 for each of the MAGNA elements are listed in Table 7.2.1; for most

nonlinear solutions, the CPU time requirement can be estimated using the formula

$$\text{CPU time} = (\text{CPU Time Factor}) \times (\text{Number of Elements}) \times (\text{Number of Integration Points/Element}) \times (\text{Number of Increments})$$

where the CPU time factor is read from the Table. Additional processing time (20-25%) should be added for additional calculations, including solution of equations. When equilibrium iterations are used in nonlinear analysis, each cycle of iteration should be counted as an "increment" when estimating computer time requirements. Since iteration cycles consume less time than an initial incremental solution step, the resulting estimates will generally be quite conservative.

Computation times for nonlinear dynamic analysis are only slightly higher than for nonlinear static solutions, and the above estimation procedure can be used with confidence. It should be noted that, in elastic-plastic analysis, the computing time cannot be estimated as accurately, since the amount of calculation per element may vary considerably. For materially nonlinear analysis, it is suggested that 30-50% overhead be added to the basic time estimate obtained from the Table.

For linear analysis with MAGNA, estimation of the required computer resources is much more difficult since CPU times are dominated by the equation solving step. Values of the CPU time factor given in the Table reflect element calculation times only; in linear analysis, assembling and solving the system of equations consumes an equal (or slightly greater) amount of processing time.

TABLE 7.2.1

COMPUTING TIME FACTORS FOR INDIVIDUAL ELEMENT TYPES (CRAY-1)

Element Type	CPU Second/Integration Point/Increment	
	Linear	Nonlinear
1	0.002	0.021
2	0.001	0.002
3	0.001	0.001
4	0.001	0.001
5	0.002	0.005
6	0.002	0.011
7	0.002	0.011
8	0.002	0.007
9	0.001	0.001
10	0.001	0.001
11	0.002	0.007
12	0.001	0.003

7.4 IBM PROGRAM VERSION (OS/VS2 MVS OPERATING SYSTEM)

Most analyses involving the IBM version of MAGNA use two batch-oriented programs: the analysis program MAGNA, and the stress averaging program STRAVG. Use of these programs, along with the OS/VS2 job control language needed to run the programs, is described in this section. Other interactive and batch-oriented programs available with the MAGNA package are not addressed in this section. See Section 11 of this manual for a discussion of the interactive plotting programs available with MAGNA; other programs available with the MAGNA package are discussed in separate manuals.

The MAGNA program is designed to allow some degree of user modification. Procedures for modifying the MAGNA load module are described in this section. Most users will probably find the default version of MAGNA adequate for their needs, however. The STRAVG program cannot be modified by the user.

The user should find reading the following subsections sufficiently informative for performing most types of analyses: the subsection on Checkpoint Restart Capabilities of MAGNA, MAGNA Execution Examples 1 through 3, and the subsection on running STRAVG. The remainder of this section deals with some of the less-frequently used options available with MAGNA.

7.4.1 Checkpoint Restart Capabilities of MAGNA

The analysis restart capabilities in MAGNA are useful in monitoring the progress of a solution, changing the solution strategy if necessary, and safeguarding against loss of data due to abnormal job terminations. Use of the restart option is discussed in Sections 5.8 and 8.3 of this manual, and job control procedures which are specific to the IBM program version are discussed below.

If an existing restart file is to be read during a job, it is read from the file connected to the ddname NOREST. The file is read from FORTRAN unit 97. Only one old restart file

can be read during a MAGNA analysis job step. The file can be protected by read-only access, and can be resident on either disk or magnetic tape. However, if the file NOREST is on tape, that tape volume must not be used for new restart files or any other I/O operations during the MAGNA execution job step. To use the same tape volume (that is, the same tape reel) for both an old restart file and new restart files, see the procedure description for new restart files below.

New restart files are written to FORTRAN unit 98. Each restart file contains information about one increment only. Each file is terminated with a FORTRAN ENDFILE statement, followed by a FORTRAN CLOSE statement. The job control file for a MAGNA run must define a file for each checkpoint restart file to be written during the job. The ddname for a new restart file must have the format "RSTxxxx", where "xxxx" has been replaced with the number of the increment to which the file corresponds. The embedded increment number must be in an I4 format, with leading zeroes coded explicitly. For example, the new restart file to be created in a MAGNA run for increment 42 must be defined as "RST0042" in the JCL stream. It is important for the user to verify the DD statements for restart files prior to a run, since the program does not check for existence of a file until just before writing to it. This occurs after the incremental solution has been reached, so failure to provide the correct ddname for a restart file will result in loss of the stored incremental results, and an abnormal job termination will occur at that point.

Restart files are sequential, unformatted files. The files generated by nonlinear analyses can be relatively large, especially for large problems, and it is possible to create several separate restart files in a single run. Although new restart files can be written to disk, users may find it more practical to write new restart files directly to magnetic tape (particularly if system disk space is at a premium). A possible

exception to this is if the problem is a linear dynamic analysis; in that case the restart files are much smaller. Note that restart files are never needed at an interactive level with the MAGNA program package.

It is possible to write multiple restart files to the same tape volume during a single job step. Again, files NOREST and RSTxxxx should not reference the same tape volume in the same job step, though. If both the old restart file and the new files are to reside on the same tape volume, then the old restart file should be first copied from tape to disk in a prior step. That disk file should then be passed to the MAGNA execution step as file NOREST; it can be deleted at the close of the job step.

Use of restart files is illustrated in the MAGNA Execution Examples 2 and 3, described in Section 7.4.5. Example 3 demonstrates how to write multiple restart files onto the same tape volume.

7.4.2 Modification of Storage Capacity

The MAGNA finite element program allocates array storage dynamically for all matrices and internal tables whose size is problem-dependent. Analyses of rather large size can be accomplished using the default array space. However, computational and input-output efficiencies for larger problems on IBM computers can be improved dramatically by optionally allocating additional storage. For very large problems (those with about 10000 degrees of freedom or more) modification of default program capacities will be required. The storage modification procedures are described below.

The storage capacity of the FORTRAN 77 version of MAGNA is controlled by the lengths of five labeled COMMON blocks declared in the main program:

1. /BLANK/ - major arrays and internal tables, including assembled stiffness, mass or effective stiffness matrix partitions.

2. /IDENT/ - tables describing the envelope of the active nonzero coefficients in the system matrices.

3. /BLOX / - tables containing matrix partitioning data for out-of-core solutions.

4. /BLEQ / - additional partitioning data for out-of-core matrix storage.

5. /USERC/ - contains working space available for use by user-written subroutines.

Table 7.4.1 shows both the minimum and the default values of each storage block.

The sizes of COMMON areas /BLANK/ and /IDENT/ determine the in-core storage capacity of the program. The COMMON area /USERC/ contains space available to the user for data storage. The remaining blocks are directly related to limits of out-of-core storage used in the solution. In general, the most effective use of the program for large-scale analysis results from increasing the blocks /BLANK/ and /IDENT/. The default lengths of the COMMON blocks /BLOX/ and /BLEQ/ are normally sufficient for all but the largest three-dimensional problems.

The array storage declared in the COMMON block /IDENT/ must be greater than the total number of unknowns in the final system of equations (including linear constraint equations). Increasing the array storage beyond that required for a particular problem will not improve program performance, however. An insufficient length of /IDENT/ will be reported by a MAGNA error message of a format described in Section 10.10.3 of this manual. Errors will generally be reported in the "NODAL VARIABLE TABLES" section of the MAGNA output listing.

TABLE 7.4.1
DEFAULT AND MINIMUM COMMON BLOCK LENGTHS
(IBM PROGRAM VERSION)

BLOCK	DEFAULT LENGTH	MINIMUM LENGTH
/BLOCK/	75000	12000
/IDENT/	10000	100*
/BLOX /	150	150
/BLEQ /	150	150
/USERC/	20	none

*Minimum size allowed by the program. Must be greater than or equal to the total number of unknowns in the final system of equations (including linear constraint equations).

The largest array areas allocated in /BLANK/ correspond to partitions of the system stiffness (or effective stiffness) matrices. Therefore, the length of this COMMON block should correlate with the number of unknowns in the model and the density of the stiffness matrix. The larger these values in the problem, the greater should be the /BLANK/ size, if possible. Situations in which this storage block can be profitably increased from the default size include larger models (10000 degrees of freedom or more), models with very large average bandwidths, and natural frequency solutions in which a number of frequencies and modes are to be extracted. Note, however, that it is never necessary to increase the /BLANK/ size beyond the minimum size shown in Table 7.4.1.

To decide how much larger to make the /BLANK/ array, a good rule of thumb is: "More is better, if you can get away with it." However, since running a load module with arrays larger than the default size can tie up a significant portion of a system's resources, the user must use some judgement in deciding how much larger the array should be. Local restrictions on job priorities certainly should be considered in this decision. Also, to avoid requesting unnecessary system resources, array sizes should not be set higher than can possibly be used by the program. The size of /BLANK/ relates directly to the number of partitions used in the problem solution; if there is only one partition, increasing /BLANK/ will not improve program performance. Information related to the size of /BLANK/ is reported in the "MATRIX PARTITIONING DATA" section of the MAGNA output listing.

To implement changes in storage capacity on an IBM system, it is currently necessary to perform three steps:

1. Edit the FORTRAN source code of the main program.

2. Compile the new main program, and create an object module.

3. Use the linkage editor to create a new load module.

The edit changes are discussed below. The resulting source code should be compiled with the IBM VS FORTRAN compiler, using the LANGLVL(77) option. Interactive compilation is possible due to the small size of the main program, and normally is recommended for debugging purposes. Note that the program requires double precision for all real values. Linkage edit procedures are discussed in a separate section of this chapter.

Storage arrays which can be modified by the user are dimensioned with values set in a FORTRAN 77 PARAMETER statement. To change the array dimensions, the user must change the appropriate PARAMETER value in the main program from the default value to the desired new value. No other editing is required. (Note that the source code for the MAGNA main program is supplied to your installation with the MAGNA package.) The PARAMETER statement and the appropriate COMMON statements are shown in Figure 7.4.1. Default PARAMETER values are shown in Figure 7.4.2.

To change the allocation for the COMMON block /BLANK/, the user must replace n1 (as shown in the PARAMETER statement of Figure 7.4.1) with the desired value; to change /BLOX/ and /BLEQ/, n3 must be changed; and to change /IDENT/, n2 must be changed. For example, if the user wishes to set /BLANK/ to a length of 18000, the PARAMETER statement would be edited to read:

```
PARAMETER ( LBLANK =   18000,
                  .
                  .
                  .
                 etc.
```

All PARAMETER values must be set to non-zero integer constants. Note that IDBLE in the PARAMETER statement must always be 2 for IBM machines, to reflect the use of double precision words.

Note that when the program storage capacity is modified, the REGION size on the JOB and EXEC statements may have to be adjusted upward. Additional increases in the REGION size may also be necessary when user-written subroutines (see Section 9) are included in a new load module. For storage of small amounts of data defined and used in these routines, a reserved COMMON block (COMMON /USERC/) currently is provided. The default length of this block is 20 REAL*8 words. To modify this value, the PARAMETER value n4 shown in Figure 7.4.1 must be changed.

For machines with virtual memory capabilities (which include most IBM machines that can run MAGNA), the "thrashing" condition should be considered when increasing program size. This is a condition in which several concurrently executing processes compete for system resources to the point where system performance is severely degraded. The normal user of MAGNA will rarely, if ever, encounter this problem. However, if a modified load module requires storage which approaches a system's limits, the user should be aware of the possiblity of thrashing, and should attempt to monitor program performance to see if a problem occurs. If it looks like the system becomes saturated due to the program's execution, the user should reduce the storage lengths wherever possible to a point more acceptable to the system.

```
      PROGRAM MAIN
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
C
                 .
                 .
                 .
C
      PARAMETER (  LBLANK  =    n1,
     +             LIDENT  =    n2,
     +             LBLOX   =    n3,
     +             LUSER   =    n4,
     +             NUSR10  =    n5,
     +             NSYS10  =    n6,
     +             NWPR10  =    n7,
     +             NUSR21  =    n8,
     +             NWPR21  =    n9,
     +             IDBLE   =     2   )
C
                 .
                 .
                 .
C
C --- VARIABLE-LENGTH COMMON BLOCKS
C
      COMMON / BLANK / A      (LBLANK)
      COMMON / BLOX  / NSHFT  (LBLOX )
      COMMON / BLEQ  / NEQLIM (LBLOX )
      COMMON / IDENT / ID     (LIDENT)
C
      COMMON / RAF10 / MXUR10, MXSR10, MXRL10, NOUR10,
     +                 NOSY10, IPTR10 (NUSR10)
      COMMON / RAF21 / MXUR21, MXSR21, MXRL21, NOUR21,
     +                 NOSY21, IPTR21 (NUSR21)
C
      COMMON / USERC / USPACE (LUSER )
                 .
                 .
                 .
      END
```

Figure 7.4.1. PARAMETER and COMMON Statements in the MAGNA Main Program.

```
      PROGRAM MAIN
C
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
C
      .
      .
      .
C
      PARAMETER (  LBLANK  =  75000,
     +             LIDENT  =  10000,
     +             LBLOX   =    150,
     +             LUSER   =     20,
     +             NUSR10  =    300,
     +             NSYS10  =    600,
     +             NWPR10  =   2373,
     +             NUSR21  =     57,
     +             NWPR21  =    200,
     +             IDBLE   =      2  )
C
      .
      .
      .
      END
```

Figure 7.4.2. Default PARAMETER Values in the MAGNA Main Program.

7.4.3 Modification of FORTRAN Direct Access File Parameters

The MAGNA program uses two FORTRAN direct access files in each analysis. These are connected to the FORTRAN units 10 and 21, and are always scratch files. Unit 10 contains global matrix information, while unit 21 contains data curves, nodal variable tables, and other information. Unit 10 always requires more space than unit 21.

The FORTRAN 77 version of MAGNA uses the FORTRAN OPEN statement to open these files. Parameters for each file are set in the main program, and should be adequate for most analyses. For large problems, however, the user may have to redefine some or all of these parameters. MAGNA will diagnose most cases where the parameters are inadequate for a problem, and will in those cases print an error message and terminate the run. Generally errors occur during the problem setup process, so runtime costs associated with inadequate parameter sizes are usually small. If MAGNA does not flag the direct access file parameters as being too small at the beginning of a nonlinear analysis, the parameters should be adequate for all increments of the analysis. The procedure for changing parameters is described in the following paragraphs.

The format of the direct access file-opening statement is:

```
      OPEN (    UNIT =  LUNIT,
     +        STATUS = 'SCRATCH',
     +        ACCESS = 'DIRECT',
     +          FORM = 'UNFORMATTED',
     +          RECL =  MAXRCL         )
```

where LUNIT is either 10 or 21, and MAXRCL is the record length in bytes.

Three parameters are defined in the main program for each direct access file. These are NUSR10, NSYS10, and NWPR10 for the file on unit 10; and are NUSR21, NSYS21, and

NWPR21 for the file on unit 21. See Figures 7.4.1 and 7.4.2 for the source code which performs this.

NUSR10 and NUSR21 are internal record indices used by MAGNA. The program refers to internal records as "user records". NSYS10 and NSYS21 are the maximum number of actual direct access records allowed for each file. Each actual direct access file record is referred to as a "system record" by the program.

NWPR10 and NWPR21 are the number of double precision words in each fixed length system record for the respective files. Accordingly, MAXRCL used in the OPEN statement above is (NWPR10 * 8) for unit 10; MAXRCL is (NWPR21 * 8) for unit 21. See Table 7.4.2 for the default and minimum allowed direct access file parameters.

On IBM systems, FORTRAN direct access files must be preformatted somehow before they can be accessed by FORTRAN READ and WRITE commands. The DEFINE FILE statement accomplishes this under IBM FORTRAN 66 languages; this statement is not valid, however, under VS FORTRAN LANGLVL(77), and is not used in the FORTRAN 77 version of MAGNA. Currently, under Release 3 of VS FORTRAN, the OPEN statement will accomplish preformatting of the first extent of a FORTRAN direct access file (for systems running under OS/VS or CMS). Note that the OPEN statement under Release 2 of VS FORTRAN did not accomplish this.

This method of preformatting has several implications for the MAGNA user. Recall that each direct access file has a maximum number of system records available to it. MAGNA will write to each of these records in the initialization phase of a run; every record must be preformatted by the system before MAGNA attempts the first write. In the DD statement for a FORTRAN direct access file, then, the user must specify enough space in the primary allocation to accommodate the entire direct access file. For example, the primary space allocation for unit 10 should be slightly larger than (NWPR10 * 8 * NSYS10), in terms

TABLE 7.4.2
DEFAULT AND MINIMUM VALUES OF DIRECT ACCESS FILE PARAMETERS

PARAMETER (in MAIN)	Function	Default Value	Minimum Value
NUSR10	Maximum number of user records for unit 10.	300	*
NSYS10	Maximum number of system records for unit 10.	600	*
NWPR10	Maximum no. of REAL*8 words per record on unit 10.	2373 words	40 words
N. A.	Record length for unit 10 in bytes (NWPR10 * 8)	18984 bytes	320 bytes
NUSR21	Maximum number of user records for unit 21.	57	*
NSYS21	Maximum number of system records for unit 21.	200	*
NWPR21	Maximum no. of REAL*8 words per record on unit 21.	80 words	40 words
N. A.	Record length for unit 21 in bytes (NWPR21 * 8)	640 bytes	320 bytes

*Minimum depends on the problem.

of bytes, where NWPR10 and NSYS10 are parameters as described previously in this section. Note that any secondary space allocation for a direct access file will not be used and is not necessary.

If the size of either of the direct access files must be increased, it is generally more effective to increase the number of system records rather than to increase the record length. Typically, the record length for unit 10 is set at or near the local device track length. The record length for unit 21 is generally much smaller than this: the default size is 640 bytes for most systems.

If the number of system records (NSYSxx) for a direct access file is increased, then the number of user records (NUSRxx) should be increased proportionally. NUSR10 and NUSR21 need never be larger than NSYS10 and NSYS21, respectively. Typically they can be set somewhat smaller; requirements for these parameters depend on the analysis problem.

7.4.4 Creating a New Load Module

Certain options available with MAGNA require changing the MAGNA FORTRAN code. Such options include changing the storage allocations or direct access file parameters in the MAIN program, and adding or rewriting user-supplied subroutines. When any of these options is used, a new MAGNA executable file (load module) must be created. The user will then specify this load module in the //JOBLIB or //STEPLIB JCL statements when making MAGNA runs instead of specifying the default MAGNA load module. This section describes how to create a new load module.

The first step in the process is to create the new routines. Typically these will be written in the FORTRAN 77 language, although actually any FORTRAN-callable language routines will be acceptable. ASSEMBLER and FORTRAN 66 routines are OK, for example.

The new routines must then be compiled to create one or more object modules. If the VS FORTRAN compiler is used, appropriate compiler options to specify are: NOFIPS, OBJECT, LANGLVL(77), OPT(2). The default MAGNA program was compiled with the options: NOGOSTMT, NOFIPS, OBJECT, LANGLVL(77), OPT(2), NOSDUMP. The user may wish to use the SDUMP and GOSTMT compiler options for debugging purposes. Note that OPT(3) is not recommended. Note also that OPT(3) is the default optimization level on many systems, and that in that case a user must specifically invoke the OPT(2) option to use the safer level.

Once the new object modules have been created, the user must use the IBM linkage editor to create the new load module. An existing MAGNA load module must be used as a template for the new load module. It is recommended that the system default load module always be used as the old module, for reliability reasons. Once the new load module has been created, the module can be stored permanently, or can be passed to a later job step and then deleted. If the new module is to be stored, note that the default MAGNA load module takes about 1.5 megabytes of disk storage.

An example of the JCL commands necessary to create a new MAGNA load module is shown in Figure 7.4.3. The statements are explained in detail below. Note that almost all of the dsnames shown in the example will be different on your system, since they will be set by your system management personnel locally.

Statement 1: JOB statement. This is different for every system.

Statement 2: STEP1 Statement. This calls up the IBM linkage editor, which usually has the default generic name IEWL.

Statement 3: SYSPRINT. This defines the output device or queue for the linkage editor's output. The queue shown is generally the default line printer.

Statement 4: SYSLIB. This statement defines all the system libraries needed to process the new object modules. Shown in the example is the library used for routines compiled with the VS FORTRAN compiler. Note that your system may have a different dsname for this library. A quick way to tell which libraries are needed for your system is to run the FORTVCLG system proc. Use a MSGLEVEL(1,1) in the JOB statement, and a dummy or empty file for the //FORT.SYSIN statement. The job will abort, but the job log will display the libraries needed under the ddname SYSLIB in the LKED job step. In your module creation run, use these libraries exactly as they are shown in the //SYSLIB statement of the dummy run.

Statement 5: SYSLMOD. This describes the new load module. Shown here is an arbitrary dsname; it is all right to use any name acceptable to your system. Note that the file must be a partitioned data set, and that space for the PDS directory must be allocated with the data set.

Statement 6: OLDLOAD. This defines the old load module; shown is the default system load module for the MAGNA package. The dsname for this load module is probably different on your system.

Statement 7: SYSUT1. This is a scratch file used by the linkage editor.

Statement 8: SYSLIN. These are the object modules that the user has created. Shown in the example are two object modules (each of which may contain several subroutines). One or more modules may actually be declared here.

Statement 9: DD *. This is actually part of the SYSLIN definition. It includes directives to the linkage editor.

```
1. //UDRI0000 JOB xxxxxxxxxxxx,
   //            REGION=2048K
   //*
   //*
   //*  ------------------------------------------------
   //*  -   UNIVERSITY OF DAYTON RESEARCH INSTITUTE  -
   //*  -   MAGNA RELINKING EXAMPLE.        12/05/83  -
   //*  ------------------------------------------------
   //*
   //*
   //*      -CREATE MODIFIED MAGNA LOAD FILE-
   //*      ---------------------------------
   //*
   //*
2. //STEP1  EXEC PGM=IEWL,
   //              REGION=2048K,
   //              TIME=(10,0),
   // PARM='NE,NOMAP,NOXREF,LET,LIST,SIZE=(1024K,100K)'
   //*
3. //SYSPRINT DD SYSOUT=A
   //*
4. //SYSLIB   DD DSN=SYS1.VFORTLIB,DISP=SHR
   //*
5. //SYSLMOD  DD DSN=USER.MAGNAMOD.LOAD(MAGNA),
   //            UNIT=SYSTS,
   //            DISP=(NEW,CATLG),
   //            SPACE=(TRK,(120,0,3),RLSE)
   //*
6. //OLDLOAD  DD DSN=UDRI.MAGNAPAK.LOAD,
   //            UNIT=SYSTS,
   //            DISP=SHR
   //*
7. //SYSUT1   DD UNIT=SYSDA,
   //            SPACE=(TRK,(10,5))
   //*
8. //SYSLIN   DD DSN=USER.MODULE1.OBJ,
   //            UNIT=SYSTS,
   //            DISP=SHR
   //         DD DSN=USER.MODULE2.OBJ,
   //            UNIT=SYSTS,
   //            DISP=SHR
9. //         DD *
         INCLUDE OLDLOAD(MAGNA)
         ENTRY MAIN
   /*
```

Figure 7.4.3. Job Control File for MAGNA Relinking Example.

7.4.5 MAGNA Job Control Examples

Four examples of job control files for MAGNA runs are discussed in this subsection. JCL statements for MAGNA runs are generally similar from run to run. Major differences between MAGNA job control files lie typically in the allocation and use of special datasets used only with certain types of analysis options.

The job control examples are shown in Figures 7.4.4 through 7.4.8. The following paragraphs discuss in detail some of the JCL statements from those figures. The statement numbers given in the text below refer to those shown in the figures. For brevity, only the JCL statements which have not been discussed in previous examples are noted. For example, the JOB statement is substantially the same for all examples described here; it is discussed in the description of Example 1 and is not referenced in the descriptions of Examples 2, 3, and 4. Note that most dsnames shown in the examples will be different among different IBM installations.

MAGNA Example 1. (See Figure 7.4.4). This example shows how one might run a linear, single-step analysis. The load module is the default system version. An MPOST postprocessor results file is created and saved. No restart files are created or saved.

Statement 1: JOB. This statement will be different for each installation. Check with your system management personnel for the proper format.

Statement 2: JOBLIB. This describes where the MAGNA load module resides. Typically this will be stored in the same partitioned data set as the STRAVG load module. Shown here is the default version of the load module for this system.

Statement 3: COPY1. This and the following four JCL statements in the figure copy the MAGNA primary input data file to the output printed record. This procedure is recommended

as a debugging tool for most MAGNA runs. It is also valuable for archival purposes. The copy step is optional, however. See the IBM MVS Utilities Manual for information on the IEBGENER Utility.

Statement 4: SYSPRINT. This describes the output destination for system utility messages from this job step. All output from this job is being routed to the default print queue, SYSOUT=A on this system.

Statement 5: SYSUT1. This describes the dataset name and location of the MAGNA primary input data file.

Statement 6: STEP1. This is the execution directive. The REGION and TIME limits shown here should be adequate for most MAGNA analyses run on an IBM 3081.

Statement 7. FT05F001. This is the MAGNA primary input data file, passed from the previous job step. If the COPY1 procedure is not used in the job, then the information given in Statement 5 (above) should be substituted here.

Statement 8: FT06F001. This is the MAGNA printed output file, shown here being routed to the default system print queue.

Statement 9: MPOST. This job creates a new MPOST postprocessor results file, which is defined here. Some users may wish to write the file directly to magnetic tape, instead of to disk as shown here. That is an acceptable alternative to the procedure shown here, but note that the GPLOT interactive geometry plotting program requires an MPOST file as input for postprocessor plotting.

Statement 10. FT10F001. This describes a FORTRAN direct access scratch file. Shown is the space needed for the default-sized file (assuming an IBM 3350 or 3380 type disk drive is used).

Statement 11: FT21F001. This describes a FORTRAN direct access scratch file. Shown is the space needed for the default-sized file.

Statement 12: FT11F001. This describes one of the sequential scratch files used in the analysis. This file has the same attributes as the other twelve scratch files shown in this section of the job control file.

MAGNA Example 2. (See Figure 7.4.5). This example shows the JCL for the first run of a nonlinear analysis. The MAGNA primary input data file is not copied to the output print record. The load module is a user-supplied version (perhaps with storage allocation modifications made). No MPOST file is created or saved. No input restart file is declared, since this is the initial job of the analysis; output restart files are written to disk.

Statement 1: STEP1. This job runs a previously created user-modified load module for this execution step. The user has designated the modified version of the load module to have the member name "MAGNA" in the load module partitioned data set.

Statement 2: STEPLIB. This describes the user-modified load module dsname and location. The STEPLIB statement was used here in lieu of a JOBLIB statement; this is typically how a user would reference a modified load module.

Statement 3: FT05F001. In this example, the input data file was not copied to the printed output record. This statement shows how to define the input data file dsname and location for MAGNA in that case.

Statement 4: No output MPOST file. This example does not create an MPOST file, so no DD statement is necessary in the job. Note that if the primary input data file specifies the creation of an MPOST file (IOPT(11) is non-zero), a DD statement

for the new MPOST file must be made in the JCL file or an abnormal termination of the job will occur.

Statement 5: RST001. This analysis will create new checkpoint restart files. RST001 refers to the file written at the first increment. The restart files shown in this job are being written to disk.

Statement 6: RST002. The input data file has specified that restart files be written at every increment. (This is not obvious from this example; that information is contained in the MAGNA input file, which is not shown here). File RST002 is the new restart file for increment 2. Note that MAGNA does not require that restart files be written at every increment; it is generally most efficient to create restart files at a less frequent interval - say, at every third or fifth increment.

Statement 7: RST003. This is the new restart file for increment 3. Note that if the analysis calculates results for increment 4 and tries to write out a restart file, the job will terminate abnormally, since the DD statement for RST004 is not declared in the JCL file.

MAGNA Example 3. (See Figure 7.4.6). This example shows the JCL used for an intermediate run in a nonlinear analysis. The run uses an input restart file and creates two new restart files. All three restart files reside or will reside on the same magnetic tape reel. An MPOST file is created and saved.

Statement 1: SETUP. This is a JES2 request to a system operator to mount the magnetic tape with the external label UDPI03. The format required for this statement may be different at your installation.

Statement 2: COPY1. This copies the MAGNA primary input file to the printed output record.

Statement 3: COPY2. This MAGNA analysis requires an old restart file as input, and will create two new restart files in the analysis job step. The user wishes all three files to reside on the same tape volume (same tape reel). On the tape, the old restart file is the third file, and the new restart files will be the fourth and fifth files, respectively. The COPY2 step copies the old restart file to a scratch disk file. This scratch disk file will be passed to the analysis step to be read as the old restart file (ddname NOREST). This allows the tape reel to be used only for write operations in the analysis step.

Statement 4: SYSUT1. This describes the old restart file on the tape reel. Note that the "RETAIN" parameter is specified in the VOL qualifier so that the tape reel will remain mounted for the next job step.

Statement 5: SYSUT2. This is the disk file to which the old restart file is copied. This disk file is passed to the analysis job step, where it has the ddname NOREST. The file is deleted at the end of that job step.

Statement 6: NOREST. An old restart file always has the ddname NOREST. This one is passed on disk from the COPY2 step above.

Statement 7: MPOST. The analysis creates an MPOST file. This syntax shows how to refer to a file which has been preallocated. Note that the file MPOST is rewound by MAGNA prior to the initial write, so any information which was stored on the file previously will be destroyed.

Statement 8: RST005. A new restart file for increment 5 will be written to tape as the fourth file on the tape.

Statement 9: RST010. A new restart file for increment 10 will be written to tape as the fifth file on the tape.

MAGNA Example 4. (See Figures 7.4.7 and 7.4.8). This example shows a job which uses the eigenvalue-with-prestress analysis option available with MAGNA. The analysis is performed in two job steps: the first step performs a nonlinear static analysis to obtain a prestress; the second step performs an eigenvalue analysis using results from the initial prestress. The first step saves files STIFF and MPOST; the second step uses these files for input under the ddnames NPREST and MPOLD, respectively. A new (and different) MPOST file is created in the second step. The file with ddname STIFF is used as a scratch file in the second step.

Example 4, Run 1 of 2: Nonlinear Prestress Step.

Statement 1: MPOST. Deformations from the nonlinear step are stored on the MPOST file. The second job step can use this information as an optional input file, under ddname MPOLD. (The deformed geometry from the nonlinear step is used as the "reference" geometry in the frequency analysis step).

Statement 2: STIFF. This file will store the element stiffness coefficients which are generated in the nonlinear step. The file must be saved in this case in order to run the frequency analysis. In all other types of analyses, STIFF is used as a scratch file.

Example 4, Run 2 of 2: Frequency Analysis Step.

Statement 1: NPREST. This is the file created with the ddname STIFF in the nonlinear static step.

Statement 2: MPOLD. This is an optional input file. It is the file created under the ddname MPOST in the nonlinear static step.

Statement 3: MPOST. In this frequency analysis step, the statically deformed geometry is stored on this file as the "undeformed", or reference, geometry. The "displacements" on

the MPOST file are those determined by the vibration mode shapes, superimposed onto the reference geometry.

Statement 4: STIFF. Note that this file is again used as a scratch file for this job step.

7.4.6 Stress Averaging Postprocessor STRAVG

The MAGNA analysis program generates as output stress values at element integration points. Stress smoothing and extrapolation to obtain nodal values of analysis data can be performed using the STRAVG program. This postprocessing program (which is separate from the MAGNA program) is described in Sections 5.7 and 10.11 of this manual. Note that a MAGNA analysis must produce an MPOST postprocessor file for STRAVG to be used on that analysis.

STRAVG is typically run as a step in a batch run on IBM machines. Job control language for batch-type runs is described in this section. A file summary table for STRAVG is shown in Table 7.4.4. Note that STRAVG execution errors are described on page 10.11.3 of this manual.

Two examples of the use of STRAVG are shown in this section, in Figures 7.4.9 and 7.4.10. JCL statements which are prefixed by numbers in those examples are discussed below.

Stress Averaging Example 1. (See Figure 7.4.9). This example shows a run which is performed in a job separate to the corresponding MAGNA analysis. Typically the user will run STRAVG in the same job as the initial MAGNA analysis, but this is not necessary as long as the MPOST file is saved in the initial analysis job.

Statement 1: JOB. This statement will be different for each installation. Check with your system management personnel for the proper format.

Statement 2: JOBLIB. This describes where the STRAVG load module resides. Typically the load module will be

stored in the same partitioned data set as the MAGNA default load module.

Statement 3: STEPA. The default member name of the stress averager is "STRAVG".

Statement 4: FT05F001. The file connected to unit 5 contains print-file control information for STRAVG. No directives are input in this example, so STRAVG will select the default printing option: full printing. A DD DUMMY statement here should produce equivalent results. Print-file directives are described in Section 5.7 of this manual.

Statement 5: FT90F001. Unit 90 is connected to the MPOST file, which has been generated by MAGNA in a previous step. Note that this file will not be modified during this STRAVG run. DISP=SHR is recommended so the user may also access the MPOST file for other purposes, such as plotting.

Statement 6: FT06F001. Unit 6 will receive the STRAVG printed output. The example is sending the output to SYSOUT=A, which is typically the default system printer.

Statement 7: FT98F001. Unit 98 will receive the new APOST postprocessor file. This file is used by the contour plotter CPLOT as well as other postprocessing programs.

Statement 8: FT66F001. Unit 66 receives unwanted print-file output. This example specifies full printing, so unit 66 will not be used in this job. The unit should always be declared, though. Typically it will be declared DD DUMMY, even when the "PRINT=NO" directive is specified in unit 5.

Statement 9: FT10F001. This file is used as a sequential scratch file.

Statement 10: FT12F001. This file is used as a sequential scratch file.

Statement 11: FT14F001. This file is used as a sequential scratch file.

Statement 12: FT20F001. This file is used as a FORTRAN direct access scratch file.

Stress Averaging Example 2. (see Figure 7.4.10). This example shows a job which runs both MAGNA and the STRAVG. The MAGNA analysis shown in this example is that from MAGNA Example 3 (Figure 7.4.6). The stress averaging example is typical of how a user would run a step in a nonlinear analysis sequence, with restart files being read and created, an MPOST file being created, and the stress averaging program being run to create an APOST file.

Statement 1: MPOST. The output MPOST file must be created in this step in order to run STRAVG in a later step.

Statement 2: STEP2. This is the execution step for STRAVG. Note that in this job, both MAGNA and STRAVG are modules in the same partitioned data set, shown in the //JOBLIB statement at the beginning of t is JCL listing.

Statement 3: FT05F001. This defines the print-file control information for STRAVG. Shown is in-stream data. This data requests normal printed output from STRAVG, with increments 2, 4, and 12 through 14 to be processed.

Statement 4: FT90F001. The input MPOST file is passed from the previous job step.

```
1. //UDRI0000 JOB xxxxxxxxxxxx,
   //         REGION=3072K
   //*
   //*
   //*    ---------------------------------------------
   //*    -   UNIVERSITY OF DAYTON RESEARCH INSTITUTE   -
   //*    -   MAGNA EXAMPLE 1.                 12/01/83  -
   //*    ---------------------------------------------
   //*
   //*
   //*        -SET JOBLIB-
   //*        ------------
   //*
2. //JOBLIB   DD DSN=UDRI.MAGNAPAK.LOAD,
   //            UNIT=SYSTS,
   //            DISP=SHR
   //*
   //*
   //*    *********************************************
   //*    *                                           *
   //*    *        M A G N A   E X E C U T I O N      *
   //*    *                                           *
   //*    *********************************************
   //*
   //*
   //*
   //*        -COPY OUT INPUT FILE-
   //*        ---------------------
   //*
3. //COPY1   EXEC PGM=IEBGENER
4. //SYSPRINT DD SYSOUT=A
   //SYSIN    DD DUMMY
5. //SYSUT1   DD DSN=UDRI.MAGEXAM1.DATA,
   //            UNIT=SYSTS,
   //            DISP=SHR
   //SYSUT2   DD SYSOUT=A,
   //            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
   //*
   //*
   //*        -GO STEP-
   //*        ---------
   //*
6. //STEP1   EXEC PGM=MAGNA,
   //            REGION=3072K,
   //            TIME=(10,0)
   //*
```

Figure 7.4.4. Job Control File for MAGNA Example 1.

```
    //*
    //*
    //*       -INPUT DATA FILE DEFINITIONS-
    //*       -----------------------------
    //*
    //*         PRIMARY INPUT FILE
    //*         ------------------
    //*
 7. //FT05F001 DD DSN=*.COPY1.SYSUT1,
    //            DISP=SHR
    //*
    //*         INPUT RESTART FILE
    //*         ------------------
    //*
    //*         (NONE THIS RUN).
    //*
    //*
    //*       -OUTPUT FILE DEFINITIONS-
    //*       -------------------------
    //*
    //*         PRINTED OUTPUT
    //*         --------------
    //*
 8. //FT06F001 DD SYSOUT=A
    //*
    //*         OUTPUT POST-PROCESSOR FILE
    //*         --------------------------
    //*
 9. //MPOST    DD DSN=UDRI.MAGEXAM1.MPOST.DATA,
    //            DISP=(NEW,CATLG),
    //            UNIT=SYSTS,
    //            DCB=(RECFM=FB,LRECL=133,BLKSIZE=5320),
    //            SPACE=(TRK,(50,50),RLSE)
    //*
    //*
    //*         OUTPUT RESTART FILES
    //*         --------------------
    //*
    //*         (NONE THIS RUN).
    //*
    //*
```

Figure 7.4.4. Job Control File for MAGNA Example 1 (Continued).

```
    //*
    //*      -DIRECT ACCESS SCRATCH FILES-
    //*      -----------------------------
    //*
10. //FT10F001 DD DSN=&&NTK,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            SPACE=(TRK,(600,0))
11. //FT21F001 DD DSN=&&NTMS,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            SPACE=(640,(200,0))
    //*
    //*
    //*      -SEQUENTIAL SCRATCH FILES-
    //*      --------------------------
    //*
12. //FT11F001 DD DSN=&&NTVC,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //            SPACE=(TRK,(50,20),RLSE)
    //STIFF    DD DSN=&&NTEKM,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //            SPACE=(TRK,(50,20),RLSE)
    //FT13F001 DD DSN=&&NTNL,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //            SPACE=(TRK,(50,20),RLSE)
    //FT14F001 DD DSN=&&NTCON,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //            SPACE=(TRK,(50,20),RLSE)
    //FT15F001 DD DSN=&&NTEC,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //            SPACE=(TRK,(50,20),RLSE)
    //FT16F001 DD DSN=&&NTPR,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //            SPACE=(TRK,(50,20),RLSE)
    //*
```

Figure 7.4.4. Job Control File for MAGNA Example 1 (Continued).

```
//*
//FT17F001 DD DSN=&&NIPI,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT18F001 DD DSN=&&NIPO,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT20F001 DD DSN=&&NTVECT,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT22F001 DD DSN=&&NTKL,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT23F001 DD DSN=&&NTED,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT24F001 DD DSN=&&NLCR,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT50F001 DD DSN=&&NTMPS,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//*
/*
```

Figure 7.4.4. Job Control File for MAGNA Example 1 (Concluded).

```
   //UDRI0000 JOB xxxxxxxxxxxxx,
   //         REGION=3072K
   //*
   //*
   //*   ---------------------------------------------
   //*   -   UNIVERSITY OF DAYTON RESEARCH INSTITUTE   -
   //*   -   MAGNA EXAMPLE 2.                12/01/83  -
   //*   ---------------------------------------------
   //*
   //*
   //*
   //*
   //*   *********************************************
   //*   *                                           *
   //*   *       M A G N A   E X E C U T I O N       *
   //*   *                                           *
   //*   *********************************************
   //*
   //*
   //*
   //*       -GO STEP-
   //*       ---------
   //*
1. //STEP1  EXEC PGM=MAGNA,
   //            REGION=3072K,
   //            TIME=(10,0)
   //*
   //*
2. //STEPLIB  DD DSN=USER.MAGNAMOD.LOAD,
   //            UNIT=SYSTS,
   //            DISP=SHR
   //*
   //*
   //*       -INPUT DATA FILE DEFINITIONS-
   //*       -----------------------------
   //*
   //*         PRIMARY INPUT FILE
   //*         ------------------
   //*
3. //FT05F001 DD DSN=UDRI.MAGEXAM2.DATA,
   //            UNIT=SYSTS,
   //            DISP=SHR
   //*
   //*         INPUT RESTART FILE
   //*         ------------------
   //*
   //*         (NONE THIS RUN).
   //*
   //*
```

Figure 7.4.5. Job Control File for MAGNA Example 2.

```
   //*
   //*      -OUTPUT FILE DEFINITIONS-
   //*      -------------------------
   //*
   //*        PRINTED OUTPUT
   //*        --------------
   //*
   //FT06F001 DD SYSOUT=A
   //*
   //*        OUTPUT POST-PROCESSOR FILE
   //*        --------------------------
   //*
4. //*        (NONE THIS RUN).
   //*
   //*        OUTPUT RESTART FILES
   //*        --------------------
   //*
5. //RST0001  DD DSN=UDRI.MAGEXAM2.RESTART1.DATA,
   //            DISP=(NEW,CATLG),
   //            UNIT=SYSTS,
   //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
   //            SPACE=(TRK,(50,50),RLSE)
   //*
6. //RST0002  DD DSN=UDRI.MAGEXAM2.RESTART2.DATA,
   //            DISP=(NEW,CATLG),
   //            UNIT=SYSTS,
   //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
   //            SPACE=(TRK,(50,50),RLSE)
   //*
7. //RST0003  DD DSN=UDRI.MAGEXAM2.RESTART3.DATA,
   //            DISP=(NEW,CATLG),
   //            UNIT=SYSTS,
   //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
   //            SPACE=(TRK,(50,50),RLSE)
   //*
   //*
   //*      -DIRECT ACCESS SCRATCH FILES-
   //*      -----------------------------
   //*
   //FT10F001 DD DSN=&&NTK,
   //            UNIT=SYSDA,
   //            DISP=(NEW,DELETE),
   //            SPACE=(TRK,(600,0))
   //FT21F001 DD DSN=&&NTMS,
   //            UNIT=SYSDA,
   //            DISP=(NEW,DELETE),
   //            SPACE=(640,(200,0))
   //*
```

Figure 7.4.5. Job Control File for MAGNA Example 2 (Continued).

```
//*
//*
//*       -SEQUENTIAL SCRATCH FILES-
//*       --------------------------
//*
//FT11F001 DD DSN=&&NTVC,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//STIFF    DD DSN=&&NTEKM,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT13F001 DD DSN=&&NTNL,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT14F001 DD DSN=&&NTCON,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT15F001 DD DSN=&&NTEC,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT16F001 DD DSN=&&NTPR,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT17F001 DD DSN=&&NIPI,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT18F001 DD DSN=&&NIPO,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//*
```

Figure 7.4.5. Job Control File for MAGNA Example 2 (Continued).

```
//*
//FT20F001 DD DSN=&&NTVECT,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT22F001 DD DSN=&&NTKL,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT23F001 DD DSN=&&NTED,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT24F001 DD DSN=&&NLCR,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT50F001 DD DSN=&&NTMPS,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//*
/*
```

Figure 7.4.5. Job Control File for MAGNA Example 2 (Concluded).

```
         //UDRI0000 JOB xxxxxxxxxxxxx,
         //         REGION=3072K
         //*
         //*
      1. /*SETUP  PLEASE MOUNT TAPE (UDRI03)    UDRI/RING IN
         //*
         //*
         //*  ---------------------------------------------
         //*  -   UNIVERSITY OF DAYTON RESEARCH INSTITUTE  -
         //*  -   MAGNA EXAMPLE 3.                12/01/83 -
         //*  ---------------------------------------------
         //*
         //*
         //*      -SET JOBLIB-
         //*      ------------
         //*
         //JOBLIB   DD DSN=UDRI.MAGNAPAK.LOAD,
         //            UNIT=SYSTS,
         //            DISP=SHR
         //*
         //*
         //*  *********************************************
         //*  *                                           *
         //*  *       M A G N A   E X E C U T I O N       *
         //*  *                                           *
         //*  *********************************************
         //*
         //*
         //*
         //*      -COPY OUT INPUT FILE-
         //*      ---------------------
         //*
      2. //COPY1  EXEC PGM=IEBGENER
         //SYSPRINT DD SYSOUT=A
         //SYSIN    DD DUMMY
         //SYSUT1   DD DSN=UDRI.MAGEXAM3.DATA,
         //            UNIT=SYSTS,
         //            DISP=SHR
         //SYSUT2   DD SYSOUT=A,
         //            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
         //*
         //*
```

Figure 7.4.6. Job Control File for MAGNA Example 3.

```
    //*
    //*        -COPY INPUT RESTART FILE TO SCRATCH DISC-
    //*        -----------------------------------------
    //*
 3. //COPY2  EXEC PGM=IEBGENER
    //SYSPRINT DD SYSOUT=A
    //SYSIN    DD DUMMY
 4. //SYSUT1   DD DSN=TAPFILE3,
    //            UNIT=TAPE,
    //            DISP=OLD,
    //            DCB=(DEN=4),
    //            VOL=(,RETAIN,SER=UDRI03),
    //            LABEL=(3,NL)
 5. //SYSUT2   DD DSN=OLDREST,
    //            UNIT=SYSDA,
    //            DISP=(NEW,PASS),
    //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //            SPACE=(TRK,(50,50),RLSE)
    //*
    //*
    //*        -GO STEP-
    //*        ---------
    //*
    //STEP1  EXEC PGM=MAGNA,
    //            REGION=3072K,
    //            TIME=(10,0)
    //*
    //*
    //*        -INPUT DATA FILE DEFINITIONS-
    //*        -----------------------------
    //*
    //*          PRIMARY INPUT FILE
    //*          ------------------
    //*
    //FT05F001 DD DSN=*.COPY1.SYSUT1,
    //            DISP=SHR
    //*
    //*          INPUT RESTART FILE
    //*          ------------------
    //*
 6. //NOREST   DD DSN=*.COPY2.SYSUT2,
    //            DISP=(OLD,DELETE)
    //*
    //*
```

Figure 7.4.6. Job Control File for MAGNA Example 3 (Continued).

```
    //*
    //*       -OUTPUT FILE DEFINITIONS-
    //*       -------------------------
    //*
    //*         PRINTED OUTPUT
    //*         --------------
    //*
    //FT06F001 DD SYSOUT=A
    //*
    //*         OUTPUT POST-PROCESSOR FILE
    //*         --------------------------
    //*
 7. //MPOST    DD DSN=UDRI.MAGEXAM3.MPOST.DATA,
    //            UNIT=SYSTS,
    //            DISP=MOD
    //*
    //*         OUTPUT RESTART FILES
    //*         --------------------
    //*
 8. //RST0005  DD DSN=UDRI.MAGEXAM3.REST05.DATA,
    //            UNIT=TAPE,
    //            DISP=(NEW,KEEP),
    //            DCB=(DEN=4,RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //            VOL=(,RETAIN,SER=UDRI03),
    //            LABEL=(4,NL)
    //*
 9. //RST0010  DD DSN=UDRI.MAGEXAM3.REST10.DATA,
    //            UNIT=TAPE,
    //            DISP=(NEW,KEEP),
    //            DCB=(DEN=4,RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //            VOL=(,RETAIN,SER=UDRI03),
    //            LABEL=(5,NL)
    //*
    //*
    //*       -DIRECT ACCESS SCRATCH FILES-
    //*       -----------------------------
    //*
    //FT10F001 DD DSN=&&NTK,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            SPACE=(TRK,(600,0))
    //FT21F001 DD DSN=&&NTMS,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            SPACE=(640,(200,0))
    //*
    //*
```

Figure 7.4.6. Job Control File for MAGNA Example 3 (Continued).

```
//*
//*      -SEQUENTIAL SCRATCH FILES-
//*      -------------------------
//*
//FT11F001 DD DSN=&&NTVC,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//STIFF    DD DSN=&&NTEKM,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//FT13F001 DD DSN=&&NTNL,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//FT14F001 DD DSN=&&NTCON,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//FT15F001 DD DSN=&&NTEC,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//FT16F001 DD DSN=&&NTPR,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//FT17F001 DD DSN=&&NIPI,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//FT18F001 DD DSN=&&NIPO,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//*
```

Figure 7.4.6. Job Control File for MAGNA Example 3 (Continued).

```
//*
//FT20F001 DD DSN=&&NTVECT,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT22F001 DD DSN=&&NTKL,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT23F001 DD DSN=&&NTED,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT24F001 DD DSN=&&NLCR,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT50F001 DD DSN=&&NTMPS,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//*
/*
```

Figure 7.4.6. Job Control File for MAGNA Example 3 (Concluded).

```
//UDRI0000 JOB xxxxxxxxxxxxx,
//         REGION=3072K
//*
//*
//*  ----------------------------------------------
//*  -   UNIVERSITY OF DAYTON RESEARCH INSTITUTE  -
//*  -   MAGNA EXAMPLE 4.                 12/01/83  -
//*  -   RUN 1 OF 2.    NONLINEAR PRESTRESS STEP  -
//*  ----------------------------------------------
//*
//*
//*      -SET JOBLIB-
//*      ------------
//*
//JOBLIB   DD DSN=UDRI.MAGNAPAK.LOAD,
//            UNIT=SYSTS,
//            DISP=SHR
//*
//*
//*  **********************************************
//*  *                                            *
//*  *       M A G N A   E X E C U T I O N        *
//*  *                                            *
//*  **********************************************
//*
//*
//*
//*      -COPY OUT INPUT FILE-
//*      ---------------------
//*
//COPY1  EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN    DD DUMMY
//SYSUT1   DD DSN=UDRI.MAGEX4A.DATA,
//            UNIT=SYSTS,
//            DISP=SHR
//SYSUT2   DD SYSOUT=A,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//*
//*
//*      -GO STEP-
//*      ---------
//*
//STEP1  EXEC PGM=MAGNA,
//            REGION=3072K,
//            TIME=(10,0)
//*
```

Figure 7.4.7. Job Control File for MAGNA Example 4, Run 1 of 2.

```
//*
//*
//*       -INPUT DATA FILE DEFINITIONS-
//*       -----------------------------
//*
//*         PRIMARY INPUT FILE
//*         ------------------
//*
//FT05F001 DD DSN=*.COPY1.SYSUT1,
//             DISP=SHR
//*
//*         INPUT RESTART FILE
//*         ------------------
//*
//*         (NONE THIS RUN).
//*
//*
//*       -OUTPUT FILE DEFINITIONS-
//*       -------------------------
//*
//*         PRINTED OUTPUT
//*         --------------
//*
//FT06F001 DD SYSOUT=A
//*
//*         OUTPUT POST-PROCESSOR FILE
//*         --------------------------
//*
1. //MPOST    DD DSN=UDRI.MAGEX4A.NLMPOST.DATA,
//             DISP=(NEW,CATLG),
//             UNIT=SYSTS,
//             DCB=(RECFM=FB,LRECL=133,BLKSIZE=5320),
//             SPACE=(TRK,(50,50),RLSE)
//*
//*
//*         OUTPUT NONLINEAR STIFFNESS FILE
//*         -------------------------------
//*
2. //STIFF    DD DSN=UDRI.MAGEX4A.NLSTIFF.DATA
//             UNIT=SYSTS,
//             DISP=(NEW,CATLG),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//*
```

Figure 7.4.7. Job Control File for MAGNA Example 4, Run 1 of 2 (Continued).

```
//*
//*
//*          OUTPUT RESTART FILES
//*          --------------------
//*
//*          (NONE THIS RUN).
//*
//*
//*        -DIRECT ACCESS SCRATCH FILES-
//*        -----------------------------
//*
//FT10F001 DD DSN=&&NTK,
//              UNIT=SYSDA,
//              DISP=(NEW,DELETE),
//              SPACE=(TRK,(600,0))
//FT21F001 DD DSN=&&NTMS,
//              UNIT=SYSDA,
//              DISP=(NEW,DELETE),
//              SPACE=(640,(200,0))
//*
//*
//*        -SEQUENTIAL SCRATCH FILES-
//*        --------------------------
//*
//FT11F001 DD DSN=&&NTVC,
//              UNIT=SYSDA,
//              DISP=(NEW,DELETE),
//              DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//              SPACE=(TRK,(50,20),RLSE)
//FT13F001 DD DSN=&&NTNL,
//              UNIT=SYSDA,
//              DISP=(NEW,DELETE),
//              DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//              SPACE=(TRK,(50,20),RLSE)
//FT14F001 DD DSN=&&NTCON,
//              UNIT=SYSDA,
//              DISP=(NEW,DELETE),
//              DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//              SPACE=(TRK,(50,20),RLSE)
//FT15F001 DD DSN=&&NTEC,
//              UNIT=SYSDA,
//              DISP=(NEW,DELETE),
//              DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//              SPACE=(TRK,(50,20),RLSE)
//*
```

Figure 7.4.7. Job Control File for MAGNA Example 4, Run 1 of 2 (Continued).

```
//*
//FT16F001 DD DSN=&&NTPR,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT17F001 DD DSN=&&NIPI,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT18F001 DD DSN=&&NIPO,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT20F001 DD DSN=&&NTVECT,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT22F001 DD DSN=&&NTKL,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT23F001 DD DSN=&&NTED,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT24F001 DD DSN=&&NLCR,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT50F001 DD DSN=&&NTMPS,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//*
/*
```

Figure 7.4.7. Job Control File for MAGNA Example 4, Run 1 of 2 (Concluded).

```
//UDRI0000 JOB xxxxxxxxxxxxx,
//          REGION=3072K
//*
//*
//*  -----------------------------------------------
//*  -   UNIVERSITY OF DAYTON RESEARCH INSTITUTE   -
//*  -   MAGNA EXAMPLE 4.                12/01/83  -
//*  -   RUN 2 OF 2.     FREQUENCY ANALYSIS STEP   -
//*  -----------------------------------------------
//*
//*
//*      -SET JOBLIB-
//*      ------------
//*
//JOBLIB   DD DSN=UDRI.MAGNAPAK.LOAD,
//            UNIT=SYSTS,
//            DISP=SHR
//*
//*
//*  ***********************************************
//*  *                                             *
//*  *        M A G N A   E X E C U T I O N        *
//*  *                                             *
//*  ***********************************************
//*
//*
//*
//*      -COPY OUT INPUT FILE-
//*      ---------------------
//*
//COPY1  EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN    DD DUMMY
//SYSUT1   DD DSN=UDRI.MAGEX4B.DATA,
//            UNIT=SYSTS,
//            DISP=SHR
//SYSUT2   DD SYSOUT=A,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//*
//*
//*      -GO STEP-
//*      ---------
//*
//STEP1  EXEC PGM=MAGNA,
//            REGION=3072K,
//            TIME=(10,0)
//*
```

Figure 7.4.8. Job Control File for MAGNA Example 4, Run 2 of 2.

```
   //*
   //*
   //*       -INPUT DATA FILE DEFINITIONS-
   //*       -----------------------------
   //*
   //*         PRIMARY INPUT FILE
   //*         ------------------
   //*
   //FT05F001 DD DSN=*.COPY1.SYSUT1,
   //            DISP=SHR
   //*
   //*         INPUT NONLINEAR STIFFNESS FILE
   //*         ------------------------------
   //*
1. //NPREST   DD DSN=UDRI.MAGEX4A.NLSTIFF.DATA,
   //            UNIT=SYSTS,
   //            DISP=SHR
   //*
   //*         INPUT NONLINEAR GEOMETRY FILE
   //*         -----------------------------
   //*
2. //MPOLD    DD DSN=UDRI.MAGEX4A.NLMPOST.DATA,
   //            UNIT=SYSTS,
   //            DISP=SHR
   //*
   //*         INPUT RESTART FILE
   //*         ------------------
   //*
   //*         (NONE THIS RUN).
   //*
   //*
   //*       -OUTPUT FILE DEFINITIONS-
   //*       -------------------------
   //*
   //*         PRINTED OUTPUT
   //*         --------------
   //*
   //FT06F001 DD SYSOUT=A
   //*
   //*         OUTPUT POST-PROCESSOR FILE
   //*         --------------------------
   //*
3. //MPOST    DD DSN=UDRI.MAGEX4B.FRQMPOST.DATA,
   //            DISP=(NEW,CATLG),
   //            UNIT=SYSTS,
   //            DCB=(RECFM=FB,LRECL=133,BLKSIZE=5320),
   //            SPACE=(TRK,(50,50),RLSE)
   //*
```

Figure 7.4.8. Job Control File for MAGNA Example 4, Run 2 of 2 (Continued).

```
    //*
    //*        OUTPUT RESTART FILES
    //*        --------------------
    //*
    //*        (NONE THIS RUN).
    //*
    //*
    //*       -DIRECT ACCESS SCRATCH FILES-
    //*       -----------------------------
    //*
    //FT10F001 DD DSN=&&NTK,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            SPACE=(TRK,(600,0))
    //FT21F001 DD DSN=&&NTMS,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            SPACE=(640,(200,0))
    //*
    //*
    //*       -SEQUENTIAL SCRATCH FILES-
    //*       --------------------------
    //*
    //FT11F001 DD DSN=&&NTVC,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //            SPACE=(TRK,(50,20),RLSE)
 4. //STIFF    DD DSN=&&NTEKM,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //            SPACE=(TRK,(50,20),RLSE)
    //FT13F001 DD DSN=&&NTNL,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //            SPACE=(TRK,(50,20),RLSE)
    //FT14F001 DD DSN=&&NTCON,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //            SPACE=(TRK,(50,20),RLSE)
    //FT15F001 DD DSN=&&NTEC,
    //            UNIT=SYSDA,
    //            DISP=(NEW,DELETE),
    //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //            SPACE=(TRK,(50,20),RLSE)
    //*
```

Figure 7.4.8. Job Control File for MAGNA Example 4, Run 2 of 2 (Continued).

```
//*
//FT16F001 DD DSN=&&NTPR,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT17F001 DD DSN=&&NIPI,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT18F001 DD DSN=&&NIPO,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT20F001 DD DSN=&&NTVECT,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT22F001 DD DSN=&&NTKL,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT23F001 DD DSN=&&NTED,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT24F001 DD DSN=&&NLCR,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//FT50F001 DD DSN=&&NTMPS,
//             UNIT=SYSDA,
//             DISP=(NEW,DELETE),
//             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//             SPACE=(TRK,(50,20),RLSE)
//*
/*
```

Figure 7.4.8. Job Control File for MAGNA Example 4, Run 2 of 2 (Concluded).

TABLE 7.4.3
MAGNA FILE SUMMARY TABLE.

DDNAME	File Status	File Attributes and Comments for MAGNA Analysis Program
FT05F001	Old	FORTRAN Unit: 5 JCL Definition: Required. DSNAME: User-defined. Formatted, sequential access, non-indexed. DCB: RECFM=FB, LRECL=80, BLKSIZE=3200. Maximum record length: 80 bytes. Primary input data file for MAGNA analysis. See Section 8 of this manual for file description and format information.
FT06F001	New	FORTRAN Unit: 6 JCL Definition: Required. DSNAME: Normally DD SYSOUT=A. Formatted, sequential access, non-indexed. If saved on disc, use DCB: RECFM=FB, LRECL=133, BLKSIZE=5985. Maximum record length: 133 bytes, including carriage control. MAGNA printed output file. Print options can be specified using unit 5.
FT10F001	Scratch	FORTRAN Unit: 10 JCL Definition: Required. DSNAME: &&NTK. Unformatted, FORTRAN direct access. Opened using FORTRAN 77 OPEN statement. Default record length: 18984 bytes. Default maximum no. of records: 600. MAGNA scratch file. Default values shown above can be reset by the user by changing the appropriate PARAMETER in the MAIN program. Minimum allowed record length for this file is 320 bytes.

TABLE 7.4.3 (CONTINUED)
MAGNA FILE SUMMARY TABLE.

DDNAME	File Status	File Attributes and Comments for MAGNA Analysis Program
FT11F001	Scratch	FORTRAN Unit: 11 JCL Definition: Required. DSNAME: &&NTVC.
FT13F001	Scratch	Fortran Unit: 13 JCL Definition: Required. DSNAME: &&NTNL.
FT14F001	Scratch	FORTRAN Unit: 14 JCL Definition: Required. DSNAME: &&NTCON.
FT15F001	Scratch	FORTRAN Unit: 15 JCL Definition: Required. DSNAME: &&NTEC.
FT16F001	Scratch	FORTRAN Unit: 16 JCL Definition: Required. DSNAME: &&NTPR.
FT17F001	Scratch	FORTRAN Unit: 17 JCL Definition: Required. DSNAME: &&NIPI.
FT18F001	Scratch	FORTRAN Unit: 18 JCL Definition: Required. DSNAME: &&NIPO.
FT20F001	Scratch	FORTRAN Unit: 20 JCL Definition: Required. DSNAME: &&NTVECT. MAGNA sequential scratch files. Unformatted, sequential access, non-indexed. DCB: RECFM=VBS, LRECL=X, BLKSIZE=19069. Maximum record length depends on the problem size.

TABLE 7.4.3 (CONTINUED)
MAGNA FILE SUMMARY TABLE.

DDNAME	File Status	File Attributes and Comments for MAGNA Analysis Program
STIFF	Scratch or New or Mod	FORTRAN Unit: 12 JCL Definition: Required. DSNAME: &&NTEKM or user-defined. Unformatted, sequential access, non-indexed. DCB: RECFM=VBS, LRECL=X, BLKSIZE=19069. Maximum record length is 26568 bytes. File STIFF contains element stiffness information. Generally it is a scratch file. The file must be saved, however, in the first step (nonlinear step) of an analysis using the Natural Frequency Analysis with Prestress Effects option. See Sections 4.5, 5.9, 7.4, and 8.3 of this manual for discussions of this option. See also comments for file NPREST below.
FT21F001	Scratch	FORTRAN Unit: 21 JCL Definition: Required. DSNAME: &&NTMS. Unformatted, FORTRAN direct access. Opened using FORTRAN 77 OPEN statement. Default record length: 640 bytes. Default maximum no. of records: 200. MAGNA scratch file. Default values shown above can be reset by the user by changing the appropriate PARAMETER in the MAIN program. Minimum allowed record length for this file is 320 bytes.
FT22F001	Scratch	FORTRAN Unit: 22 JCL Definition: Required. DSNAME: &&NTKL.
FT23F001	Scratch	FORTRAN Unit: 23 JCL Definition: Required. DSNAME: &&NTED. MAGNA sequential scratch files. Unformatted, sequential access, non-indexed. DCB: RECFM=VBS, LRECL=X, BLKSIZE=19069. Maximum record length depends on the problem size.

TABLE 7.4.3 (CONTINUED)
MAGNA FILE SUMMARY TABLE.

DDNAME	File Status	File Attributes and Comments for MAGNA Analysis Program
FT24F001	Scratch	FORTRAN Unit: 24 JCL Definition: Required. DSNAME: &&NLCR.
FT50F001	Scratch	FORTRAN Unit: 50 JCL Definition: Required. DSNAME: &&NTMPS. MAGNA sequential scratch files. Unformatted, sequential access, non-indexed. DCB: RECFM=VBS, LRECL=X, BLKSIZE=19069. Maximum record length depends on the problem size.
NPREST	Old	FORTRAN Unit: 55 JCL Definition: Optional. DSNAME: User-defined. Unformatted, sequential access, non-indexed. DCB: RECFM=VBS, LRECL=X, BLKSIZE=19069. Maximum record length is 26568 bytes. This file is required only when the Natural Frequency Analysis with Prestress Effects option is selected. In that case, the file must be declared in the second step (natural frequency analysis step) of the analysis. It is not declared in the first step. In the first step (nonlinear analysis step) of the option, the file STIFF must be saved. That file is then input in the second step under this DDNAME, NPREST. The file is used for input only; no modification of the connected file will take place during the job step. See Sections 4.5, 5.9, 7.4, and 8.3 of this manual for discussions of the Prestress Effects analysis option.

TABLE 7.4.3 (CONTINUED)
MAGNA FILE SUMMARY TABLE.

DDNAME	File Status	File Attributes and Comments for MAGNA Analysis Program
MPOST	New or Mod	FORTRAN Unit: 90 JCL Definition: Optional. DSNAME: User-defined. Formatted, sequential, non-indexed. DCB: RECFM=FB, LRECL=133, BLKSIZE=5985. Maximum record length: 133 bytes, including carriage control. MAGNA output postprocessor results file. Incremental data output to this file is controlled using the MAGNA primary input data file, unit 5 above. All requested increments are written to a single file during a single job execution. See Section 5.7 of this manual for file structure and format.
MPOLD	Old	FORTRAN Unit: 92 JCL Definition: Optional. DSNAME: User-defined. Formatted, sequential, non-indexed. DCB: RECFM=FB, LRECL=133, BLKSIZE=5985. Maximum record length: 133 bytes, including carriage control. This file is used only in the second step of an analysis in which the Natural Frequency Analysis with Prestress Effects option has been selected. In that case, it is optional, depending on user-declared input options selected in the MAGNA primary input data file. The file connected to MPOLD in the second step is the old MPOST file saved during the first step (nonlinear step) of the analysis. That file contains the reference geometry for the natural frequency analysis step. The output MPOST file from the second step of the analysis (requested in the MAGNA primary input file and written to file MPOST as usual) will then contain (a) the prestressed state geometry as the "undeformed" geometry, and (b) the vibration modes superimposed onto the prestressed state geometry in the "displacement" sections of the file. For further information on the Natural Frequency with Prestress Effects analysis option, see Sections 4.5, 5.9, 7.4, and 8.3 of this manual.

TABLE 7.4.3 (CONCLUDED)
MAGNA FILE SUMMARY TABLE.

DDNAME	File Status	File Attributes and Comments for MAGNA Analysis Program
NOREST	Old	FORTRAN Unit: 97 JCL Definition: Optional. DSNAME: User-defined. Unformatted, sequential, non-indexed. DCB: RECFM=VBS, LRECL=X, BLKSIZE=19069. Maximum record length depends on problem size. Input checkpoint restart file. This file is used for input only; no modification of the connected file will take place during the job step. Checkpoint restarts are controlled by the MAGNA primary input data file. See this section for a discussion of restarts from analysis checkpoints.
RSTxxxx	New or Mod	FORTRAN Unit: 98 JCL Definition: Optional. DSNAME: User-defined. Unformatted, sequential, non-indexed. DCB: RECFM=VBS, LRECL=X, BLKSIZE=19069. Maximum record length depends on problem size. Output checkpoint restart file(s). Each checkpoint increment requested is written to a separate file; each file is terminated using a FORTRAN ENDFILE statement. In the DDNAME, the characters "xxxx" must be replaced with the number of the increment to which the file corresponds. The increment number must be in an I4 format, with zeroes coded explicitly. For example, the new output restart file for increment 42 must be defined as "RST0042" in the JCL file. A file must be declared in the JCL stream for each checkpoint increment to be written during the job step. See this section for a discussion of the MAGNA checkpoint restart capabilities. Note that restart files from nonlinear runs may be relatively large. Since these files are never used interactively, it may be advantageous for the user to either write the file directly to tape, or to write the files to scratch disc, and copy those files to tape in a job step following the MAGNA execution.

```
1. //UDRI0000 JOB xxxxxxxxxxxxx,
   //         REGION=2048K
   //*
   //*
   //*   ----------------------------------------------
   //*   -   UNIVERSITY OF DAYTON RESEARCH INSTITUTE   -
   //*   -   STRESS AVERAGING EXAMPLE 1.     11/29/83  -
   //*   ----------------------------------------------
   //*
   //*
   //*       -SET JOBLIB-
   //*       ------------
   //*
2. //JOBLIB   DD DSN=UDRI.MAGNAPAK.LOAD,
   //            UNIT=SYSTS,
   //            DISP=SHR
   //*
   //*
   //*   **********************************************
   //*   *                                            *
   //*   *        S T R E S S   A V E R A G I N G     *
   //*   *                                            *
   //*   **********************************************
   //*
   //*
3. //STEPA  EXEC PGM=STRAVG,
   //            REGION=2048K,
   //            TIME=(20,0)
   //*
   //*
   //*       -FILE DEFINITIONS FOR STRESS AVERAGER-
   //*       --------------------------------------
   //*
   //*
   //*         -INPUT FILES-
   //*         -------------
   //*
4. //FT05F001 DD DATA
   /*
   //*
5. //FT90F001 DD DSN=UDRI.STREXAM1.MPOST.DATA,
   //            UNIT=SYSTS,
   //            DISP=SHR
   //*
```

Figure 7.4.9. Job Control File for STRAVG Example 1.

```
         //*
         //*         -OUTPUT FILES-
         //*         --------------
         //*
      6. //FT06F001 DD SYSOUT=A
         //*
      7. //FT98F001 DD DSN=UDRI.EXAMPLE1.APOST.DATA,
         //            DISP=(NEW,CATLG),
         //            UNIT=SYSTS,
         //            DCB=(RECFM=FB,LRECL=133,BLKSIZE=5320),
         //            SPACE=(TRK,(50,50),RLSE)
         //*
         //*
         //*         -SCRATCH FILES-
         //*         ---------------
         //*
      8. //FT66F001 DD DUMMY
         //*
      9. //FT10F001 DD DSN=&&IEFILE,
         //            UNIT=SYSDA,
         //            DISP=(NEW,DELETE),
         //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
         //            SPACE=(TRK,(50,20),RLSE)
         //*
     10. //FT12F001 DD DSN=&&IGFILE,
         //            UNIT=SYSDA,
         //            DISP=(NEW,DELETE),
         //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
         //            SPACE=(TRK,(50,20),RLSE)
         //*
     11. //FT14F001 DD DSN=&&IDFILE,
         //            UNIT=SYSDA,
         //            DISP=(NEW,DELETE),
         //            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
         //            SPACE=(TRK,(50,20),RLSE)
         //*
     12. //FT20F001 DD DSN=&&IRAND,
         //            DISP=(NEW,DELETE),
         //            UNIT=SYSDA,
         //            SPACE=(TRK,(50,0))
         //*
         /*
```

Figure 7.4.9. Job Control File for STRAVG Example 1 (Concluded).

```
//UDRI0000 JOB xxxxxxxxxxxx,
//         REGION=3072K
//*
//*
/*SETUP  PLEASE MOUNT TAPE (UDRI03)    UDRI/RING IN
//*
//*
//*   ---------------------------------------------
//*   -   UNIVERSITY OF DAYTON RESEARCH INSTITUTE  -
//*   -   STRESS AVERAGING EXAMPLE 2.    12/01/83  -
//*   ---------------------------------------------
//*
//*
//*       -SET JOBLIB-
//*       ------------
//*
//JOBLIB   DD DSN=UDRI.MAGNAPAK.LOAD,
//            UNIT=SYSTS,
//            DISP=SHR
//*
//*
//*   **********************************************
//*   *                                            *
//*   *       M A G N A   E X E C U T I O N        *
//*   *                                            *
//*   **********************************************
//*
//*
//*
//*       -COPY OUT INPUT FILE-
//*       ---------------------
//*
//COPY1  EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN    DD DUMMY
//SYSUT1   DD DSN=UDRI.MAGEXAM3.DATA,
//            UNIT=SYSTS,
//            DISP=SHR
//SYSUT2   DD SYSOUT=A,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//*
//*
```

Figure 7.4.10. Job Control File for STRAVG Example 2.

```
//*
//*       -COPY INPUT RESTART FILE TO SCRATCH DISC-
//*       -----------------------------------------
//*
//COPY2  EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN    DD DUMMY
//SYSUT1   DD DSN=TAPFILE3,
//            UNIT=TAPE,
//            DISP=OLD,
//            DCB=(DEN=4),
//            VOL=(,RETAIN,SER=UDRIO3),
//            LABEL=(3,NL)
//SYSUT2   DD DSN=OLDREST,
//            UNIT=SYSDA,
//            DISP=(NEW,PASS),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,50),RLSE)
//*
//*
//*       -GO STEP-
//*       ---------
//*
//STEP1  EXEC PGM=MAGNA,
//            REGION=3072K,
//            TIME=(10,C)
//*
//*
//*       -INPUT DATA FILE DEFINITIONS-
//*       -----------------------------
//*
//*         PRIMARY INPUT FILE
//*         ------------------
//*
//FT05F001 DD DSN=*.COPY1.SYSUT1,
//            DISP=SHR
//*
//*         INPUT RESTART FILE
//*         ------------------
//*
//NOREST   DD DSN=*.COPY2.SYSUT2,
//            DISP=(OLD,DELETE)
//*
//*
//*       -OUTPUT FILE DEFINITIONS-
//*       -------------------------
//*
//*         PRINTED OUTPUT
//*         --------------
//*
```

Figure 7.4.10. Job Control File for STRAVG Example 2 (Continued).

```
    //*
    //FT06F001 DD SYSOUT=A
    //*
    //*         OUTPUT POST-PROCESSOR FILE
    //*         -------------------------
    //*
 1. //MPOST     DD DSN=UDRI.MAGEXAM3.MPOST.DATA,
    //             UNIT=SYSTS,
    //             DISP=MOD
    //*
    //*         OUTPUT RESTART FILES
    //*         -------------------
    //*
    //RST0005   DD DSN=UDRI.MAGEXAM3.REST05.DATA,
    //             UNIT=TAPE,
    //             DISP=(NEW,KEEP),
    //             DCB=(DEN=4,RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //             VOL=(,RETAIN,SER=UDRI03),
    //             LABEL=(4,NL)
    //*
    //RST0010   DD DSN=UDRI.MAGEXAM3.REST10.DATA,
    //             UNIT=TAPE,
    //             DISP=(NEW,KEEP),
    //             DCB=(DEN=4,RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //             VOL=(,RETAIN,SER=UDRI03),
    //             LABEL=(5,NL)
    //*
    //*
    //*       -DIRECT ACCESS SCRATCH FILES-
    //*       -----------------------------
    //*
    //FT10F001 DD DSN=&&NTK,
    //             UNIT=SYSDA,
    //             DISP=(NEW,DELETE),
    //             SPACE=(TRK,(600,0))
    //FT21F001 DD DSN=&&NTMS,
    //             UNIT=SYSDA,
    //             DISP=(NEW,DELETE),
    //             SPACE=(640,(200,0))
    //*
    //*
    //*       -SEQUENTIAL SCRATCH FILES-
    //*       --------------------------
    //*
    //FT11F001 DD DSN=&&NTVC,
    //             UNIT=SYSDA,
    //             DISP=(NEW,DELETE),
    //             DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
    //             SPACE=(TRK,(50,20),RLSE)
```

Figure 7.4.10. Job Control File for STRAVG Example 2 (Continued).

```
//*
//STIFF    DD DSN=&&NTEKM,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//FT13F001 DD DSN=&&NTNL,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//FT14F001 DD DSN=&&NTCON,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//FT15F001 DD DSN=&&NTEC,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//FT16F001 DD DSN=&&NTPR,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//FT17F001 DD DSN=&&NIPI,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//FT18F001 DD DSN=&&NIPO,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//FT20F001 DD DSN=&&NTVECT,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//FT22F001 DD DSN=&&NTKL,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//*
```

Figure 7.4.10. Job Control File for STRAVG Example 2 (Continued).

```
   //*
   //FT23F001 DD DSN=&&NTED,
   //              UNIT=SYSDA,
   //              DISP=(NEW,DELETE),
   //              DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
   //              SPACE=(TRK,(50,20),RLSE)
   //FT24F001 DD DSN=&&NLCR,
   //              UNIT=SYSDA,
   //              DISP=(NEW,DELETE),
   //              DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
   //              SPACE=(TRK,(50,20),RLSE)
   //FT50F001 DD DSN=&&NTMPS,
   //              UNIT=SYSDA,
   //              DISP=(NEW,DELETE),
   //              DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
   //              SPACE=(TRK,(50,20),RLSE)
   //*
   //*
   //*
   //*   **********************************************
   //*   *                                            *
   //*   *       S T R E S S   A V E R A G I N G      *
   //*   *                                            *
   //*   **********************************************
   //*
   //*
2. //STEP2   EXEC PGM=STRAVG,
   //              REGION=2048K,
   //              TIME=(10,0)
   //*
   //*
   //*      -FILE DEFINITIONS FOR STRESS AVERAGER-
   //*      --------------------------------------
   //*
   //*
   //*        -INPUT FILES-
   //*        -------------
   //*
3. //FT05F001 DD DATA
   PRINT=YES
       2    4
      12  -15
   /*
   //*
4. //FT90F001 DD DSN=*.STEP1.MPOST,
   //              DISP=SHR
   //*
```

Figure 7.4.10. Job Control File for STRAVG Example 2 (Continued).

```
//*
//*
//*       -OUTPUT FILES-
//*       --------------
//*
//FT06F001 DD SYSOUT=A
//*
//FT98F001 DD DSN=UDRI.EXAMPLE2.APOST.DATA,
//            UNIT=SYSTS,
//            DISP=MOD
//*
//*       -SCRATCH FILES-
//*       ---------------
//*
//FT66F001 DD DUMMY
//*
//FT10F001 DD DSN=&&IEFILE,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//*
//FT12F001 DD DSN=&&IGFILE,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//*
//FT14F001 DD DSN=&&IDFILE,
//            UNIT=SYSDA,
//            DISP=(NEW,DELETE),
//            DCB=(RECFM=VBS,LRECL=X,BLKSIZE=19069),
//            SPACE=(TRK,(50,20),RLSE)
//*
//FT20F001 DD DSN=&&IRAND,
//            DISP=(NEW,DELETE),
//            UNIT=SYSDA,
//            SPACE=(TRK,(50,0))
//*
/*
```

Figure 7.4.10. Job Control File for STRAVG Example 2 (Concluded).

TABLE 7.4.4
STRAVG FILE SUMMARY TABLE.

DDNAME	File Status	File Attributes and Comments for STRAVG Stress Averaging Program
FT05F001	Old	FORTRAN Unit: 5 JCL Definition: Required. DSNAME: User-defined. Formatted, sequential access, non-indexed. DCB: RECFM=FB, LRECL=80, BLKSIZE=3200. Maximum record length: 80 bytes. Print-file control information; processing control information. The file may be empty (data definition DD DUMMY is OK) but it must be declared. It is generally short enough to be included in the JCL stream. See Section 5.7 of this manual for a file description and format information.
FT06F001	New or Mod	FORTRAN Unit: 6 JCL Definition: Required. DSNAME: Normally DD SYSOUT=A. Formatted, sequential access, non-indexed. If saved on disc, use DCB: RECFM=FB, LRECL=133, BLKSIZE=5985. Maximum record length: 133 bytes, including carriage control. STRAVG printed output file. Print options can be specified using unit 5.
FT10F001	Scratch	FORTRAN Unit 10 JCL Definition: Required. DSNAME: &&IEFILE.
FT12F001	Scratch	FORTRAN Unit 12 JCL Definition: Required. DSNAME: &&IGFILE.
FT14F001	Scratch	FORTRAN Unit: 14 JCL defintion: Required. DSNAME: &&IDFILE STRAVG sequential scratch files. Unformatted, sequential access, non-indexed. DCB: RECFM=VBS, LRECL=X, BLKSIZE=19069. Maximum record length depends on the problem size.

TABLE 7.4.4 (CONCLUDED)
STRAVG FILE SUMMARY TABLE.

DDNAME	File Status	File Attributes and Comments for STRAVG Stress Averaging Program
FT20F001	Scratch	FORTRAN Unit: 20 JCL Definition: Required. DSNAME: &&IRAND. Unformatted, FORTRAN direct access. Opened using the FORTRAN statement: DEFINE FILE 20 (50, 12680, L, IVRAND). The parameters on this direct access file cannot be changed by the user.
FT66F001	Scratch	FORTRAN Unit: 66 JCL Definition: Required. DSNAME: Generallly DD DUMMY. Unwanted print-file output is shunted to this unit if the user selects the "PRINT=NO" option. If the file is to be saved, use the same file attributes used with unit 6.
FT90F001	Old	FORTRAN Unit: 90. JCL Definition: Required. DSNAME: User-defined. Formatted, sequential, non-indexed. DCB: RECFM=FB, LRECL=133, BLKSIZE=5985. Maximum record length: 133 bytes, including carriage control. Old MPOST file. This is the main input data file for STRAVG. It can be used with read-only access.
FT98F001	New or Mod	FORTRAN Unit: 98 JCL definition: Required. DSNAME: User-defined. Formatted, sequential, non-indexed. DCB: RECFM=FB, LRECL=133, BLKSIZE=5985. Maximum record length: 133 bytes, including carriage control. New APOST file. This is the main output file from the STRAVG program. It is meant to be used with CPLOT and other post-processing programs. The file can be relatively large; it is approximately the same size as the MPOST file in many cases. See Section 5.7 of this manual for information about file structure and format.

CARD	COL	DATA	DESCRIPTION	NOTES
1 (cont)	37-40	IOPT(10)	Flag for User Loads Subroutine =0: Normal Loads Input =1: User Subroutine(s) Provided	-
	41-44	IOPT(11)	Flag for Postprocessor File Option =0: No Postprocessor File Written $\geq$1: Postprocessor File is to be Written on Local File MPOST at every IOPT(11)th Increment	-
	45-48	IOPT(12)	Number of Time Increment Changes in Nonlinear Solution	(6)
	49-52	IOPT(13)	Variable Time Step Flag =0: Fixed Time or Loading Increments =1: Automatic Variable Time Step to be Used in Solution	(7)
	53-56	IOPT(14)	Thermal Stress Analysis Flag =0: Neglect Thermal Effects =1: Include Thermal Effects	(8)
	57-60	IOPT(15)	Contact Analysis Flag =0: No Surface Contact =1: Include Surface Contact Analysis	(9)
	61-64	IOPT(16)	Not Used	
	65-68	IOPT(17)	Cyclic Symmetry Flag =0: No Cyclic Symmetry =1: Inactive =2: Cyclically Symmetric Structure	(10)

CARD	COL	DATA	DESCRIPTION	NOTES
2	1-4	NSTEP	Number of Solution Time Steps	(11)
	5-8	IPRF	Printing Frequency (in Increments)	(12)
	9-12	NRANGE	Number of Nodal Ranges for Printed Output (Default=Print All Nodes)	(13)
	13-16	IVPRNT	Velocity Printing Flag =0: Velocity Output Suppressed =1: Print Velocities	
	17-20	ISTART(1)	Beginning Node Number for Output Range No. 1	(13)
	21-24	IEND(1)	Final Node Number for Output Range No. 1	(13)
	25-28	ISTART(2)	Beginning Node For Range No. 2	(13)
	29-32	IEND(2)	Final Node for Range No. 2	(13)
	⋮	⋮	⋮ ⋮ ⋮	
	73-76	ISTART(8)	Beginning Node for Range No. 8	(13)
	77-80	IEND(8)	Final Node for Range No. 8	(13)
3	1-10	DT	Time (or Load Parameter) Step Size	(14)
	11-20	TZERO	Time at Start of Solution	(14)
	21-30	TMAX	Maximum Time Value	(14)
	31-40	DTMIN	Minimum Time Step Size (Variable Time-Step Option Only)	(15)
	41-50	DTMAX	Maximum Time Step Size (Variable Time-Step Option Only)	(15)
4	1-10	ALPHA	Time Integration Parameter,	(16)
	11-20	DELTA	Time Integration Parameter,	(16)

CARD	COL	DATA	DESCRIPTION	NOTES
5	1-10	BETA	Stiffness Matrix Coefficient for Proportional Damping,	(17)
	11-20	GAMMA	Mass Matrix Coefficient for Proportional Damping,	(17)
6*	1-5	INCR(1)	Increment Number for First Time Increment Change	(18)
	6-15	TIME(1)	Time Increment Value	
	16-20	INCR(2)	Increment Number for Second Time Increment Change	
	21-30	TIME(2)	Time Increment Value	
	31-35	INCR(3)	Increment Number for Third Time Increment Change	
	36-45	TIME(3)	Time Increment Value	
	46-50	INCR(4)	Increment Number for Fourth Time Increment Change	
	51-60	TIME(4)	Time Increment Value	

*Note: Card 6 is required only if IOPT(12)>0; that is, if the solution time increment is to be modified during a nonlinear analysis.

(8) IOPT(14) must be set to 1 to perform thermal stress analysis (linear or nonlinear). If the input data contains nodal temperatures, thermal expansion coefficients, etc., these data can be suppressed if desired by setting IOPT(14) = 0.

(9) Contact between three-dimensional surfaces may be considered in nonlinear static and dynamic analysis only. Contact analysis data is entered as shown in Section 8.6 if IOPT(15) > 0. At present, surface contact and thermal stress may not be analyzed simultaneously.

(10) Cyclic symmetry may only be used for natural frequency analysis at present. Part or all of the internal degrees of freedom may optionally be eliminated. Cyclic symmetry natural frequency analysis is entered if IOPT(2) = 3 and IOPT(17) = 2. Data must be supplied by the user to indicate which nodes lie on the respective symmetric boundaries and which internal nodes are to be retained and eliminated (Section 8.7).

(11) NSTEP is applicable in all nonlinear and/or transient dynamic solutions. The analysis is continued until NSTEP increments have been performed, or until the time value exceeds TMAX (Card 3). In a restart run, NSTEP is interpreted as the last increment to be performed (including those completed in the preceding run(s)).

(12) Solutions will be printed every IPRF time or load steps. It should be noted that, in linear dynamic analysis, stress and strain calculations are performed only when output is required, so that use of a small printing frequency can significantly increase solution time. Choice of an output frequency in nonlinear analysis affects solution time only slightly.

(13) If NRANGE is zero or blank, nodal solution quantities (e.g., displacements, velocities, reactions) will be output for all nodes in the model. When NRANGE > 0,

output will be generated for all nodes contained in any of the point ranges ISTART(i) - IEND(i), for i=1,2,...,NRANGE. Up to eight ranges of nodes may be specified for selective nodal output.

(14) Solutions are performed at times (TZERO + I*DT), for I=1,2,...,NSTEP, or until the value of time exceeds TMAX. In a transient dynamic solution, DT and TZERO represent actual values of time. For nonlinear static analysis, "time" is used as a load parameter; that is, all loads are specified in the form P = P(t) (see Section 8.9), and values of t_i = TZERO + I*DT are used to determine the successive load levels at which increments of the solution begin and end. As an example, a 1000 lb. load applied in increments of 100 lb. could be specified using DT = 1.0, TZERO = 0.0, and specifying (in Section 8.9) that P(t) = 100.t.

(15) DTMIN and DTMAX are used as limiting time (or load parameter) step sizes in solutions based upon the variable time step option (IOPT(13)=1). In dynamic analysis, it is advisable to specify at least a maximum time step (DTMAX), as the automatic time step feature may, in some cases, select a time step which is too large for high accuracy. In plasticity and/or contact analysis, the time step may tend to become quite small; in these circumstances, a minimum step size should always be specified.

(16) Parameters α, δ are relevant for transient dynamic analysis by Newmark's operator (implicit integration). A linear dynamic solution is unconditionally stable provided $\delta \geq 1/2$ and $\alpha \geq (1/4)\ (1/2 + \delta)^2$. Default values are $\alpha = 1/4$, $\delta = 1/2$ ("constant-average-acceleration" operator).

(17) In transient dynamic analyses, a damping matrix is permitted, of the form $C = \beta K + \gamma M$ (Rayleigh damping). For an undamped solution, $\beta = \gamma = 0$.

(18) IOPT(12) determines the number of pairs (increment, time step) to be entered in this section. The initial time step, DT, is entered on card 3 (see note 13); at increment INCR(1), the time step will be changed to the value of TIME(1), and so on. Increment values INCR(1) must be entered in ascending order. If IOPT(12) $\geq$ 0, do not enter this line of data. Note that time step changes are permitted only in nonlinear analysis, and that a maximum of four such changes is permitted.

8.3.2 Eigenvalue Solution Control Parameters

(Required only for natural frequency analyses)

A. GENERAL EIGENSOLUTION PARAMETER CARD

CARD	COL	DATA	DESCRIPTION	NOTES
1	1-5	NTRAIL	Number of Iteration Trial Vectors (Default = 2, Maximum = 50)	(1)
	6-10	NREQD	Number of Natural Frequencies to be Determined (Default=1)	(2)
	11-15	MAXIT	Maximum Number of Iterations	(3)
	16-25	TOLVEC	Vector Tolerance for Convergence of Frequency Solution (Default = 0.001)	(4)
	26-30	MFLAG	Mass Matrix Type =0: Consistent Mass =1: Lumped Mass	
	31-35	ITYPE	Frequency Range Flag =0: Lowest =1: Highest	(5)
	36-40	IPREST	Flag for Prestress Effects =0: Flag for Prestress Effects =1: Include Nonlinear Prestress Effect	(6)
	41-45	INCPRE	Flag for Initial Geometry File =0: No Initial Geometry File >0: Increment Number on Initial Geometry File Corresponding to Nonlinear, Prestressed State	(7)
	46-55	ESHIFT	Eigenvalue Shift	(8)

CARD	COL	DATA	DESCRIPTION	NOTES
1 (cont)	56-60	NUMSEG	Number of Cyclically Symmetric Substructure Segments	(9)
	61	IHARSW	Input Harmonic Selection and Data Recover Control Information Switch =Blank: Do NOT Read Recovery Data =R : Read Recovery Data	(10)
	62-65	NOHARM	Number of Cyclic Harmonics	(11)

B. CYCLIC SYMMETRIC EIGENSOLUTION OUTPUT CONTROL DATA

(Required only for natural frequency cyclically symmeteric analyses)

- Omit these cards unless IHARSW = R on the type 1 general Eigenvalue solution parameter

- Repeat this card NOHARM times if required

CARD	COL	DATA	DESCRIPTION	NOTES
2	1-5	LHARM	Cyclic Harmonic Number	(12)
	6-10	NOMODE	Number of Modes for which Natural Frequencies will be Computed	(13)
	11-15	NOSEG	Number of Segments for which Output will be Printed	(14)
	20	JPTD	Displacement Printout Flag (Default = Print Displacements) =Blank: Print Displacements =N : Do NOT print displacements	(15)
	25	JPTS	Stress, Strain, and Energy Printout Flag (Default = Print Stresses, Stains, and Energy) =Blank: Print Stresses, Strains, or Energy = N : Do NOT Print Stesses, Strains, or Energy	(16)
	30	JPPD	Displacement Postprocessor File Flag (Default = Write Displacements) =Blank: Write Displacements = N : Do NOT Write Displacements	(17)
	35	JPPS	Stress, Strain, and Energy Postprocessor Flag (Default = Write Stresses, Strains, and Energy) =Blank: Write Stresses, Strains, or Energy = N : Do NOT write Stresses, Strains, or Energy	(18)

(9) The number of cyclically symmetric segments occuring in a full revolution. The angle computed using the number of segments is used to determine conformity of the respective nodes on the two symmetric boundaries.

(10) The program by default computes the first natural frequency and mode shape for cyclic harmonics 0 through NUMSEG/2 if NUMSEG is even, or (NUMSEG-1)/2 if NUMSEG is odd, unless NOHARM > 0. If NOHARM > 0, the first mode is computed for cyclic harmonics 0 through NOHARM. By default the program prints mode shape data for the first cyclic symmetric segment and outputs displacement data to the postprocessor file. The program limits the number of cyclic harmonics for which natural frequency analyses can be performed to 25 per run. If the default number of cyclic harmonics is greater than 25, analyses are performed for cyclic harmonics 0 through 24. Stress, strain, and energy are not computed by default. If an R is entered for IHARSW, data is read to control cyclic harmonic selection, the number of modes to be computed for the respective harmonics, the computation of stress, strain, and energy, and the output of data.

(11) This parameter controls the number of different cyclic harmonics for which natural frequencies and mode shapes will be computed. If IHARSW is not equal to R, the first frequency and mode shape for harmonics 0 through NOHARM will be computed.

(12) The program requires that the cyclic harmonics be selected in ascending order. The values may range from 0 to NUMSEG/2 if NUMSEG is even, or (NUMSEG-1)/2 if NUMSEG is odd. When a value of R is entered for IHARSW program defaults are changed. Displacement, stress, strain, and energy data for the NOSEG segments specified on the previous card are printed by default. The displacement, stress, strain, and energy data are written to the postprocessor file by default. The printed and postprocessor output can be suppressed by entering an N in the appropriate fields of this card.

(13) Number of natural frequencies and mode shapes to be computed for the cyclic harmonic are specified on this card. There is no program limit on the number of modes.

(14) This parameter specifies the number of cyclic symmetric segments for which output will be printed for the cyclic harmonic specified on this card. Data is printed for the specified number of segments in ascending order starting with the first segment.

(15) An N in this field supresses printing of the mode shape data.

(16) An N in this field supresses printing of stress, strain, and energy data.

(17) An N in this field supresses the writing of displacement data to the postprocessor file.

(18) An N in this field supresses the writing of stress, stain, and energy data to the postprocessor file.

8.3.3 Analysis Restart Data (All machine versions)

(Required for all nonlinear or dynamic analyses which read or write a restart file.)

CARD	COL	DATA	DESCRIPTION	NOTES
1	1-7	-	Literal "RESTART"	
	8-10	-	(blank)	
	11-15	IREAD	Analysis Restart Flag =0: New Analysis (no Restart) =1: Read Restart File from Previous Analysis	(1)
	16	-	(blank)	
	17-20	IDOLD	Restart File Label	(2)
	21-25	INCOLD	Increment at Which Analysis is be Restarted	
	26-30	IWRITE	Checkpoint Flag =0: No Restart File to be Written =1: Restart File to be Written During Current Job	(3)
	31		(blank)	
	32-35	IDNEW	Label for New Restart File	(4)
	36-40	IRFREQ	Frequency in Number of Increments for Checkpoint Output to New Restart File	(5)

This Section removed.

This Section removed.

8.7.1 Cyclic Symmetric Partition Assignment Data

(Required only for the analysis of Cyclically Symmetric Structures)

The nodes of a cyclically symmetric substruture must be assigned by the user to two or more of four partitions. The four partitions are:

Partition 1 - Omitted internal nodes
Partition 2 - Retained internal nodes
Partition 3 - External Symmetric Boundary 1 nodes
Partition 4 - External Symmetric Boundary 2 nodes

Nodes in the omitted internal partitions are eliminated prior to analysis to reduce the number of equations in the solution set. An internal node is any node not on one of the cyclically symmetric conformable boundaries. Boundary partitions 3 and 4 are required for cyclically symmetric analyses. Partitions 1 or 2 (or both) will exist whenever there are nodes that do not lie on the symmetric boundaries.

All nodes are initially assumed to be retained internal nodes (partition 2). Omitted internal nodes (partition 1) must be identified explicitly by the user. If a node is assigned more than once to different partitions, the last assignment is used. The order in which the nodes are assigned and/or are reassigned to partitions can (depending on subsequent data - Sec 8.7.2) affect the order in which their respective degrees of freedom are numbered.

To assign nodes to partitions the user can specify nodes individually, specify a range of nodes with an optional increment, or assign all the nodes to a particular partition. The last option may be desirable if the majority of the nodes are to be assigned to a partition other than partition 2. This could occur if most or all of the internal nodes are to be omitted. Input of cyclic symmetry partition assignment data is terminated with a single blank card.

- Partition assignment data input is terminated by a single blank card (i. e., N1 = 0).

CARD	COL	DATA	DESCRIPTION	NOTES
1-n	1	RANGE	Range, list, or all nodes flag (Default = list) = L: List = R: A range of nodes = A: Assign all nodes	(1)
	2-5	IPART	Partition number (1, 2, 3, or 4)	(2)
	6-10	N1	1st Node or first node in a range	(3)
	11-15	N2	2nd Node or upper limit of a node range	
	16-20	N3	3rd Node or increment of a range (The default range increment = 1)	
	21-25	N4	4th Node	
	.	.	. .	
	.	.	. .	
	.	.	. .	
	76-80	N20	20th Node	

(1) It is assumed that the card contains a simple list of nodes to be assigned to the partition specified in the card unless column 1 contains an A or an R. If column 1 contains an A, all the nodes in the cyclic symmetric substructure are assigned to the specified partition. If column 1 contains an R, Nodes N1, N1 + N3, N1 + 2*N3, ..., N1 + n*N3 are assigned to the specified partition where n is the largest integer which results in a node number less that or equal than N2. The default value for N3 is 1.

(2) The only valid partition numbers are 1, 2, 3, 4. The default partition number 2 is assumed if the partition number is left blank or an invalid value is given.

(3) This must be a nonzero positive integer (A value of N1 less than or equal to zero will terminate reading of these cards).

8.7.2 Degree of Freedom Order Data

(Required only for the analysis of cyclically symmetric substructures)

The user can control the order in which the degrees of freedom for the respective nodes are numbered by several methods. The program requires a list of global node numbers for each of the partitions in the order that their respective degrees of freedom are to be numbered.

Nodes are not renumbered. The lists are only used to specify the order in which the degrees of freedom are numbered. The lists can be generated according to global node number, from list data supplied by the user, or sort key data associated with the nodes. The sort key data can be either generated by the program or supplied by the user.

The order in which the degrees of freedom are numbered determines the skyline profiles of the stiffness and mass matrix partitions. Computation time is a function of the average height of the skyline. This is discussed in general in chapter 12 with respect to node numbering. While cyclic symmetry (with or without omitted nodes), involves a variety of matrix operations (Sec 4.7), the guidelines given in Chapter 12 apply.

Please note that in the analysis of cyclic symmetric substructures the partition node lists (however they are generated), not the global node numbers, control how the degrees of freedom are numbered and therefore determine the matrix profiles.

The degrees of freedom for the corresponding nodes on cyclically symmetric boundaries 1 and 2 must be numbered in the same order or the symmetric boundary conditions will not be applied correctly. The program performs a geometry check to determine if this condition has been met and issues a fatal error if the two boundaries are not numbered in compatible order.

The order in which the degrees of freedom are numbered becomes much less important whenever internal degrees of freedom are omitted as all matrices except those involved in the

reduction process become full matrices. The omission of internal nodes will not result in a reduction in the use of computer resources for most analyses unless a large percentage of the internal nodes are omitted. The omission of only a small percentage of the internal notes is apt to result in a sharp increase in the use of computer resources.

By default the program first generates a complete list of the nodes included in the substructure. The node numbers of nodes included in the substructure appear in their global node number position in the default list. The list positions of any node not included in the substructure are set to zero. (i. e. position N in the default list is set to N if node N is connected to the substructure; if node N is not connected, position N is set to 0.)

The substructure node list is later rearranged to generate lists for each of the partitions. The respective order of the node numbers in the partition lists is the same as it was in the overall list.

The user can alter the default overall node list or replace with his own list before the program rearranges the list by partition. Optionally, the user can request that the program generate the partitions lists from nodal sort key data assigned to the nodes.

The method used by the program to establish the partition node number lists which determine the order in which the degrees of freedom are numbered is controlled by input data in the following card.

CARD	COL	DATA	DESCRIPTION	NOTES
1	5	IREORD	Degree of Freedom number ordering switch = 0: Use the global node numbers = 1: Use order assigned to partition and input a sort key list. = 2: Input a node list = 3: Input a sort key list = 4: Input node list data by location = 5: Input sort key data by location	(1) (2) (2)
1	6-10	NORD1	First node for which reordering data is to be read (Default = 1)	(3)

NOTE: If IREORD = 0, no further reordering data is required. If IREORD > 0, node reordering data must be entered in Sections 8.7.2A or 8.7.2B below.

NOTES:

(1) The value of IREORD determines the method the program will use to order the global degrees of freedom. The effects of the defined values are given in the following table.

Value of IREORD	Degree of Freedom Numbering Proceedure
0	The degrees of freedom are numbered by partition according to ascending global node number. No degree of freedom reordering data is read.
1	Sort key data is generated by the program as the nodes are explicitly assigned to partitions. A list of additional sort key values is read for some or all of tne nodes starting with node NORD1.
2	A list of sort key data is used to replace the default global node number sort key data starting with position NORD1 in the list.
3	A new list of node numbers is read in the order that the respective degrees of freedom are to be numbered, starting with position NORD1 in the default list.
4	Node numbers and their corresponding sort key values are read.
5	Node list data is read to modify specified locations in the default list. (Note: By default node list position N is set equal to N, if node N is included in the substucture. Otherwise position N defaults to 0.)

(2) When the node lists are generated from sort key data the sort key values are assigned to the nodes in the following sequence:

First; The global node numbers are assigned as the default sort key value for all nodes connected to the substructure. All connected nodes are considered to be initially assigned to the default partition. Nodes not connected to any partition are assigned zero sort key values.

Second; The user may request that new sort key values be assigned as the nodes are explicitly assigned to partitions. The new values are assigned in ascending order starting with a value one greater than the number of nodes. The default global node number sort key value is not altered for any node unless it is explicitly assigned to a partition. The last sort key value assigned applies to any node explicitly assigned to partitions more than once.

Third; The user may enter input data to alter any or all sort key values assigned by the first and second methods above.

A. TYPE 1 LIST DEGREE OF FREEDOM ORDER DATA

(Node or sort key values supplied as a list)

- Type 1 order data cards are read only if 1 < IREORD < 3. Nonzero data items entered in the cards are inserted into the node number list or sort key in consecutive locations starting with location NORD1.
- A blank card terminates the reading of type 1 degree of freedom ordering cards (i.e., N1 = 0). It is not necessary to enter more than one value per card. Up to twenty values may be entered per line. See Note 1. Zero values are not entered into the list and the location counter is not incremented. No entries should be made in a card after the first zero or blank entry (failure to observe this rule can have unpredictable effects).

CARD	COL	DATA	DESCRIPTION	NOTES
1-n	1-5	N1	1st Node List or Sort Key Value	(1)
	6-10	N2	2nd Node List or Sort key Value	
	11-15	N3	3rd Node List or Sort Key Value	
	16-20	N4	4th Node List or Sort key Value	
	.	.	. .	
	.	.	. .	
	.	.	. .	
	76-80	N20	20th Node List or Sort Key Value	

NOTES:

(1) Only nonzero values are entered. The location counter is incremented as each nonzero value is processed. Blank or zero values are not entered into the list and the location counter is not incremented. No entries should be made in a card after the first zero or blank entry (failure to observe this rule can have unpredictable effects).

B. TYPE 2 DEGREE OF FREEDOM DATA BY LOCATION

(Node or sort key input by node list or sort key location)

- Type 2 order data are read only if IREORD $\geq$ 4. Locations and node numbers or sort key values are read in pairs. The data items are inserted into the specified node list or sort key locations. Pairs for which the location is zero or blank are ignored except for the first location.
- A blank card terminates the reading of type 2 degree of freedom ordering cards (i. e. L1 = 0). One to ten pairs may be entered per card. Only the first pair is necessary to prevent termination of degree of freedom order data input.

CARD	COL	DATA	DESCRIPTION	NOTES
1-n	1-5	L1	1st Node List or Sort Key Location	(1)
	6-10	N1	1st Node List or Sort key Value	(2)
	11-15	L2	2nd Node List or Sort Key Location	
	16-20	N2	2nd Node List or Sort key Value	
	.	.	. .	
	.	.	. .	
	.	.	. .	
	71-75	L10	10th Node List or Sort Key Location	
	76-80	N20	10th Node List or Sort Key Value	

NOTES:

(1) The position in the node list where the list entry is to made or the global node number to which the following sort key value is to be assigned.

(2) Global node number to be entered in the list or a sort key value.

NOTES:

(1) *The values NUMC and KGEN can be used to generate a series of linear constraints, when the number of terms, the nodal components, and multipliers are identical. NUMC constraints will be generated (including the one input), by incrementing each of the node numbers involved in the constraint by KGEN each time. IF NUMC = 0, only the input constraint is generated; if NUMC.GT.0 and KGEN = 0, KGEN is assigned a default value of one.*

(2) *The node number NOD(I), and direction IC(I) = 1,2,3, and multiplier XM(I) define a single term of the linear constraint. The form of the constraint equation is then*

$$\sum_{I=1}^{NTERM} XM(I) * U(I)$$

where U(I) is the global displacement degree of freedom defined at node NOD(I) in direction IC(I). Displacement degrees of freedom which appear in linear constraint equations must not *be otherwise constrained. Linear constraint relationships involving both eliminated (omitted) and retained displacement degrees of freedom in a single relationship are not permitted. This restriction applies only to the analysis of structures or substructures with reduced degrees of freedom. Linear constraint relationships involving only eliminated degrees of freedom and those involving only retained degrees of freedom are permitted.*

SECTION 7.4:

1. ------, VS FORTRAN Application Programming: Language Reference, Release 3.0, Order No. GC26-3986-3, International Business Machines Corporation, March, 1983.

2. ------, VS FORTRAN Application Programming: Guide, Release 3.0, Order No. GC26-3985-4, International Business Machines Corporation, March, 1983.

3. ------, VS FORTRAN Application Programming: Sysytem Services Reference Supplement, Release 3.0, Order No. GC26-3988-2, International Business Machines Corporation, March, 1983.

4. ------, VS FORTRAN Application Programming: Library Reference, Release 3.0, Order No. GC26-3989-2, International Business Machines Corporation, March, 1983.

5. ------, OS/VS2 MVS JCL, Release 3.8, Order No. GC28-0692-4, International Business Machines Corporation, May, 1979.

6. ------, OS/VS Linkage Editor and Loader, VS2 Release 3.8, Order No. GC26-3813-5, International Business Machines Corporation, August, 1978.

7. ------, OS/VS Message Library: Linkage Editor and Loader Messages, VS2 Release 3.7, Order No. GC38-1007-5, International Business Machines Corporation, August, 1978.

8. ------, OS/VS MVS Utilties, Release 3.7, Order No. GC26-3902-0, International Business Machines Corporation, December, 1977.

REVISED OUTPUT LISTINGS FOR SECTION 7.1

PART II

```
GET,UTILS/UN=D828130
/UTILS

 PROBLEMS WITH MISSING PROGRAM FILE (FILE NOT FOUND)
 SHOULD BE REFFERED TO  TSGT. S. ZASTROW, 255-6100.


 MAGNA PRE/POST PROCESSOR UTILITIES

   1. PRE-PROCESSORS
   2. POST-PROCESSORS

 SELECT BY NUMBER OR TYPE Q TO QUIT ? 1

 PRE-PROCESSOR UTILITIES

   1. AGRID
   2. CORGEN
   3. CREATE
   4. EXPAND
   5. GPLOT
   6. HIDLIN
   7. IJKGEN
   8. NEUTRAL
   9. PREP
  10. REFMT
  11. SPATCH
  12. SURFDIG
  13. TRNSFR
  14  FILE UTILITIES

 SELECT BY NUMBER OR TYPE Q TO QUIT ? 3


  ***************************************
  BEGIN CREATE - DATA ENTRY AND EDITING OF
  COARSE GRID FINITE ELEMENT MODEL DATA
  ***************************************

  DO YOU HAVE A PREVIOUSLY GENERATED DATA
  FILE FROM CREATE TO RE-EDIT? (Y,N).....:
? N


  **************     INITIAL NODE POINT INPUT     **************

  IN THIS PART OF THE INPUT, COORDINATE DATA IS ENTERED AT THE
  KEYBOARD FOR EACH NODE OF THE MODEL. THE DATA IS CHECKED FOR
  CONSISTENCY AND MAY BE EDITED LATER. THE MAXIMUM ALLOWABLE
  NODE POINT NUMBER IS 500.

  ENTER THE DIMENSIONALITY OF THE MESH TO
  BE DEFINED ( 2 OR 3 ) ..................
? 2
```

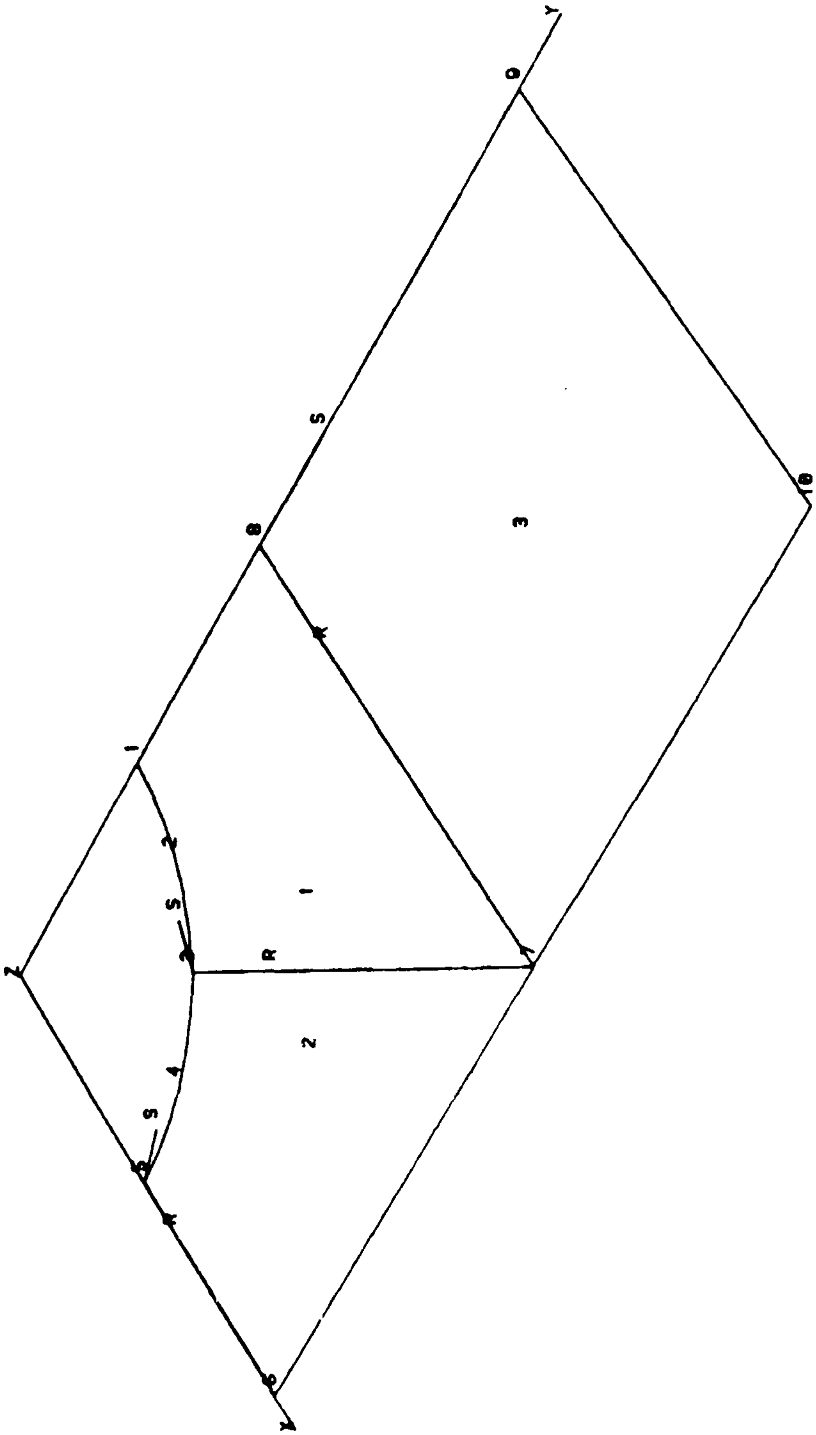
```
***** BEGIN DIRECT NODAL POINT INPUT -
ENTER NODE NO. AND 2 COORDINATES AT EACH NODE
( ENTER ALL ZEROES TO TERMINATE INPUT )

  ENTER NODE, X, Y    -
? 1   0.     5.
  ENTER NODE, X, Y    -
? 2   1.07  4.6
  ENTER NODE, X, Y    -
? 3   3.55  3.55
  ENTER NODE, X, Y    -
? 4   4.6   1.07
  ENTER NODE, X, Y    -
? 5   5.    0.
  ENTER NODE, X, Y    -
? 6  10.    0.
  ENTER NODE, X, Y    -
? 7  10.   10.
  ENTER NODE, X, Y    -
? 8   0.   10.
  ENTER NODE, X, Y    -
? 9   0.   20.
  ENTER NODE, X, Y    -
? 10 10.   20.
  ENTER NODE, X, Y    -
? 0,0,0

  EDIT THE NODAL POINT DATA ( Y , N ).....
? N

  ***************     INITIAL ELEMENT INPUT     ****************

  THIS PORTION OF THE INPUT GIVES THE CONNECTION DATA DEFINING
  FINITE ELEMENTS IN TERMS OF THE NODES. CONNECTIVITY FOR THE
  ELEMENTS FOLLOWS THE CONVENTIONS FOR TWO AND THREE DIMENSIONAL
  ELEMENTS IN MAGNA ----------

  ====== TWO-DIMENSIONAL ======  ===== THREE-DIMENSIONAL =====
  NODE POINTS     LOCATIONS         NODE POINTS    LOCATIONS
     1 - 4        CORNERS              1 -  8      CORNERS
     5 - 8        MIDSIDES             9 - 20      MIDSIDES
       9          CENTROID            21 - 26      MIDFACES
                                         27        CENTROID

  THE MAXIMUM ALLOWABLE ELEMENT NUMBER IS 100.

  FOR EACH ELEMENT, ENTER (1)ELEMENT NUMBER, (2)NUMBER LOCAL
  NODES (I.E.,THE LENGTH OF THE CONNECTIVITY LIST) AND (3)THE
  LIST OF CONNECTED NODES.  ENTER ELEMENT=0 TO TERMINATE INPUT

  FOR 2-D ELEMENTS, THE MAX. LOCAL NODE NUMBER SHOULD BE BETWEEN  2 AND  9

  ENTER ELEMENT NO.-
? 1
  MAX. LOCAL NODE  -
? 6
  CONNECTIVITY LIST-
? 8,1,3,7,0,2
```

```
  ENTER ELEMENT NO.-
? 2
  MAX. LOCAL NODE  -
? 6
  CONNECTIVITY LIST-
? 7,3,5,0,0,4

  ENTER ELEMENT NO.-
? 3
  MAX. LOCAL NODE  -
? 4
  CONNECTIVITY LIST-
? 10,0,0,7

  ENTER ELEMENT NO.-
? 0


  EDIT THE ELEMENT DATA ( Y , N )........
? Y

  ELEMENT EDITING OPTIONS - (L)IST, (I)NPUT, (D)ELETE, (E)XIT
  ENTER EDITING OPTION ( L , I , D , E )--
? L
  ENTER RANGE OF ELEMENTS TO BE LISTED ---
? 1,3

  ELMT    N1  N2  N3  N4  N5  N6  N7  N8  N9 N
     1     0   1   3   7   0   2
     2     7   3   5   6   0   4
     3    10   0   0   7

  ELEMENT EDITING OPTIONS - (L)IST, (I)NPUT, (D)ELETE, (E)XIT
  ENTER EDITING OPTION ( L , I , D , E )--
? E

  * SUMMARY OF CURRENT MODEL PARAMETERS *

  HIGHEST-NUMBERED NODE POINT........   10
  NUMBER OF UNDEFINED NODES..........    0
  HIGHEST-NUMBERED ELEMENT DEFINED...    3
  NUMBER OF UNDEFINED ELEMENTS.......    0

  NUMBER OF REFERENCES TO UNDEFINED NODES=   0


  EDIT THE NODAL POINT DATA ( Y , N ).....
? N


  EDIT THE ELEMENT DATA ( Y , N )........
? N


  ****************************************
   PREPROCESSOR DATA FILE COMPLETE
  ****************************************
```

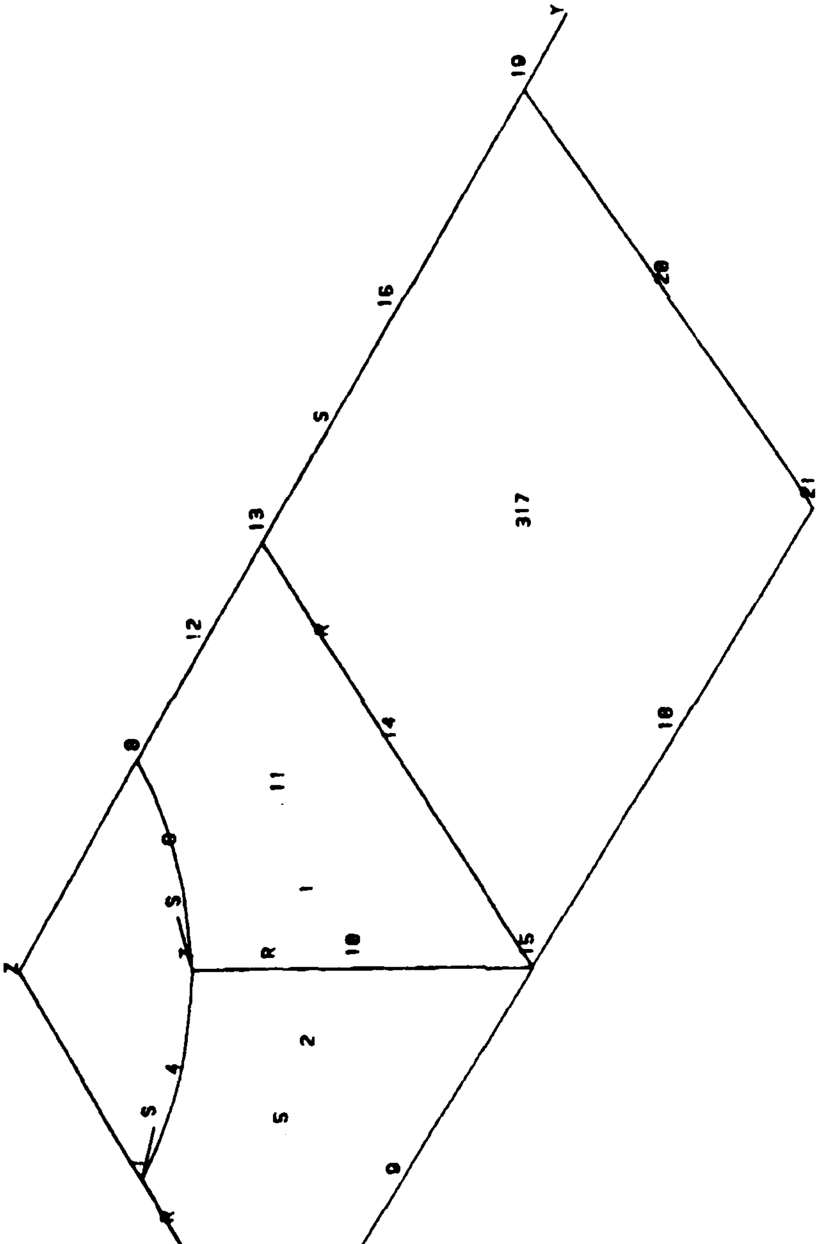
```
PRE-PROCESSOR UTILITIES

 1. AGRID
 2. CORGEN
 3. CREATE
 4. EXPAND
 5. GPLOT
 6. MIDLIN
 7. IJKGEN
 8. NEUTRAL
 9. PREP
10. REFMT
11. SPATCH
12. SURFDIG
13. TRNSFR
14. FILE UTILITIES

SELECT BY NUMBER OR TYPE Q TO QUIT ? 14

FILE PROCESSING UTILITIES

 1. ENQUIRE,F
 2. INPUT / OUTPUT FILE NAMES
 3. RENAME
 4. RETURN TO MAIN MENU

SELECT BY NUMBER OR TYPE Q TO QUIT ? 1

LOCAL FILE INFORMATION.

 FILENAME  LENGTH/PRUS  TYPE  STATUS    FS

  INPUT*          1      IN.*  EOR
  INPUT                  LO.
  OUTPUT                 LO.
  UNFMT          83      LO.   BOI
  UTILS          59      LO.   EOR
  ZZZZZC2         5      LO.   EOR WRITE NAD

 TOTAL = 6

FILE PROCESSING UTILITIES

 1. ENQUIRE,F
 2. INPUT / OUTPUT FILE NAMES
 3. RENAME
 4. RETURN TO MAIN MENU

SELECT BY NUMBER OR TYPE Q TO QUIT ? 3
ENTER PRESENT FILE NAME? UNFMT
ENTER DESIRED FILE NAME? TAPE10

FILE PROCESSING UTILITIES

 1. ENQUIRE,F
 2. INPUT / OUTPUT FILE NAMES
 3. RENAME
 4. RETURN TO MAIN MENU

SELECT BY NUMBER OR TYPE Q TO QUIT ? 4
```

```
PRE-PROCESSOR UTILITIES

  1. AGRID
  2. CORGEN
  3. CREATE
  4. EXPAND
  5. GPLOT
  6. HIDLIN
  7. IJKGEN
  8. NEUTRAL
  9. PREP
 10. REFMT
 11. SPATCH
 12. SURFDIG
 13. TRNSFR
 14. FILE UTILITIES

 SELECT BY NUMBER OR TYPE Q TO QUIT ? Q

          ****************************************
          *************  BEGIN PREP  *************
          **** MAGNA PREPROCESSING UTILITIES *****
          ****************************************


 PLOTTING TERMINAL ( Y , N ) ?::
? Y
 TEKTRONIX OR HP ( T , H ) ?  ::
? T

 TEKTRONIX TERMINAL TYPES ----
  0, 4006-1
  1, 4010 / 4012 / 4013 / 4052
  2, 4014 / 4015
  3. 4014 / 4015   ( ENH. GR. MOD. )
  4, 4114

 ENTER TERMINAL TYPE ( 0 - 4 )::
? 4
 ENTER CHARACTERS PER SECOND  ::
? 120
 NUMBER OF INPUT DATA FILES ? ::
? 1
 ENTER FILE #                 ::
? 10
 ENTER LABEL                  ::
? PLATE

 TYPE 'HELP' FOR LIST OF COMMANDS

 ::
? PLOT
 ENTER INPUT FILE #1 LABEL               ::
? PLATE

 INITIALIZATION OF PLOTTING ROUTINE ERASES SCREEN.
 READY (Y,N) ? ....:
? Y
```

```
  *
? HELP
    COMMAND         DESCRIPTION
    *******         ***********
     AXES            AXES DRAW AND LABEL
     BOUNDS          PLOT BOUNDARY CONDITIONS
     CLIP            CLIP PLANE POSITION
     CUBE            SET MINIMA AND MAXIMA
     DEFAULT         SET DEFAULT VALUES
     DRAW            DRAW MODEL
     ELEMENTS        PLOTS ALL OR SELECTED ELEMENTS
     EYE             EYE POSITION
     HELP            LIST ALL COMMANDS
     LABELS          LABELS ELEMENTS AND/OR NODES
     MAIN            RETURNS CONTROL TO MAIN PROGRAM
     ORIENT          PLOTS R,S,T ORIENTATION AXES
     PROJECTION      PROJECTION TYPE
     REFLECT         REFLECT A PLANE
     ROTATE          ROTATE MODEL ABOUT AXES
     SCALE           SCALE PLOT
     SHRINK          SHRINK ELEMENTS
     SUMMARY         LIST ALL PARAMETER VALUES
     TIME            PRINT CPU TIME SINCE START OF SESSION
     TRANSLATE       TRANSLATE MODEL FROM ORIGIN
     VERTICAL        VERTICAL AXIS
     ZOOM            ZOOM ON THE MODEL
     SITE            SITE POSITION
  *
? AXES
    PLOT AND LABEL THE AXES?(Y,N) ...............:
? Y
  *
? LABEL
    LABEL THE ELEMENTS?(Y,N) ....................:
? Y
    LABEL THE NODES?(Y,N) .......................:
? Y
    LABEL ALL THE SURFACES?(Y,N) ................:
? Y
  *
? ORIENT
    PLOT ORIENTATION AXES?(Y,N) .................:
? Y
  *
? DRAW

   AT END OF PLOT, ENTER CHARACTER TO CONTINUE.
   READY (Y,N) ? ....:
? Y
```

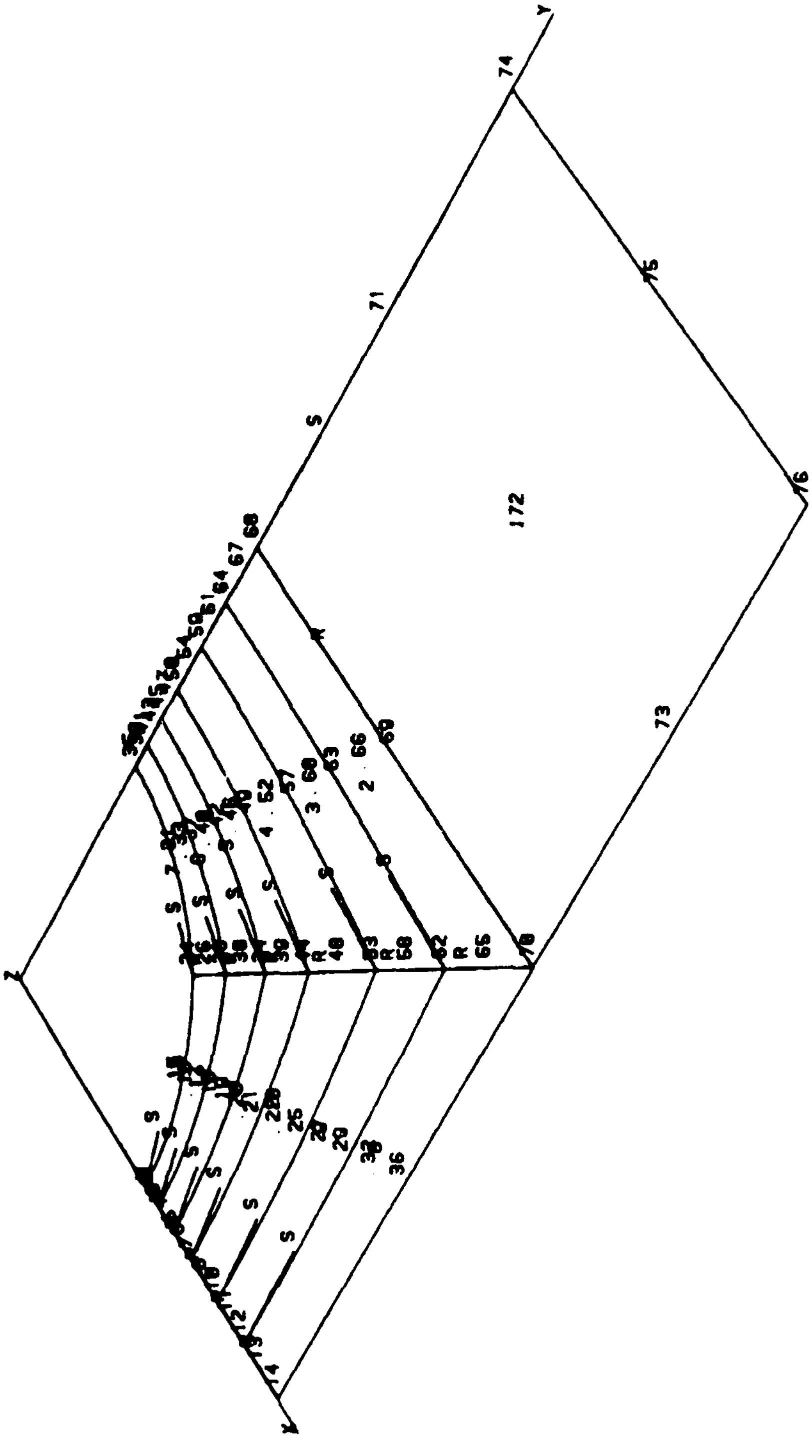


```
  *
? MAIN

 ::
? FILL
 ENTER INPUT FILE #1 LABEL                ::
? PLATE
 ENTER OUTPUT FILE LABEL                  ::
? PLAT2
 CREATING NEW DATA FILE
 FILL OPTIONS -
   1. 3-D ELEMENTS ONLY
   2. 2-D ELEMENTS ONLY
 ENTER FILL OPTION (1,2)....::
? 2
 MAX NODE POSITION # IN FINAL ELEMENTS?
   (8,9)...::
? 9

 FILL COMPLETE.
 PERFORM SUBSEQUENT TIDY ? (Y,N) ->
? Y

 TIDY OPTION ONE OR TWO ? (OPT1, OPT2, HELP) ->
? OPT2

       TIDY COMPLETE
     2 DUPLICATE NODES DELETED

 ::
? PLOT
 ENTER INPUT FILE #1 LABEL                ::
? PLAT2
  *
? DRAW

   AT END OF PLOT, ENTER CHARACTER TO CONTINUE.
   READY (Y,N) ? ....:
? Y
```



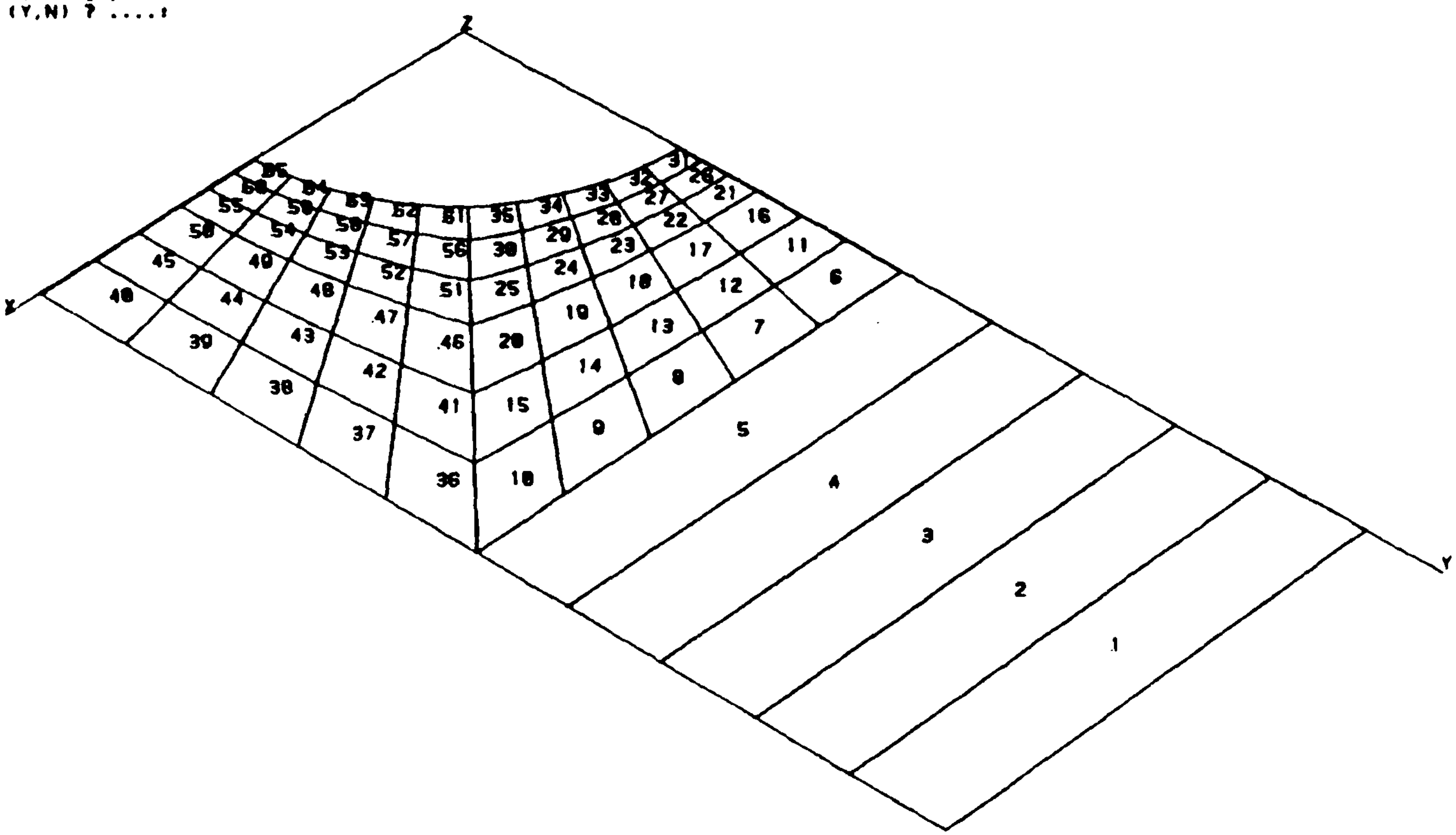
```
MAIN

 ::
? LIST

 LISTING OF DATA FILES AVAILABLE

 #10 --- PLATE
 #11 --- PLAT2

 ::
? DELE
 ENTER LABEL                ::
PLATE
?
 ::
? REFINE
 ENTER INPUT FILE #1 LABEL              ::
? PLAT2
 ENTER OUTPUT FILE LABEL                ::
? REFIN1
 CREATING NEW DATA FILE

   BEGIN REFINE ROUTINE...

 1 - RANDOM ELEMENT INPUT
 2 - RANGE OF ELEMENTS
 3 - EXECUTE REFINEMENT
 ENTER OPTION --------------->
? 1
 MAXIMUM NUMBER OF ELEMENTS
   SPECIFIABLE -------------->  150
 NUMBER STILL AVAILABLE ----->  150

 ENTER ELEMENT NUMBER AND <RETURN> FOR EACH
   ELEMENT TO BE REFINED.  ELEMENTS NEED NOT BE
   ENTERED IN ASCENDING ORDER.  ELEMENT NUMBER 0
   TERMINATES INPUT.
 ENTER ELEMENT # ------------>
? 1
 ENTER ELEMENT # ------------>
? 2
 ENTER ELEMENT # ------------>
? 0

 1 - RANDOM ELEMENT INPUT
 2 - RANGE OF ELEMENTS
 3 - EXECUTE REFINEMENT
 ENTER OPTION --------------->
? 3

 ENTER CUT DIRECTION (R,S,T) >
? R

 MAXIMUM CUTS PER ELEMENT ---> 5
 ENTER CUTS PER ELEMENT ----->
? 5

 EQUALLY SPACED CUTS ? (Y,N) >
? N
```

```
 ENTER CUT POSITIONING IN
   ASCENDING ORDER (0-100) -->
? 10 22 35 56 75

 REFINE COMPLETE.
 PERFORM SUBSEQUENT TIDY ? (Y,N) ->
? Y

 TIDY OPTION ONE OR TWO ? (OPT1, OPT2, HELP) ->
? OPT2

       TIDY COMPLETE
    11 DUPLICATE NODES DELETED

  !!
? PLOT
 ENTER INPUT FILE #1 LABEL                 !!
? REFIN1
  *
? DRAW

   AT END OF PLOT, ENTER CHARACTER TO CONTINUE.
   READY (Y,N) ? ....!
? Y
```


Y
74
71
75
76
172
73
36

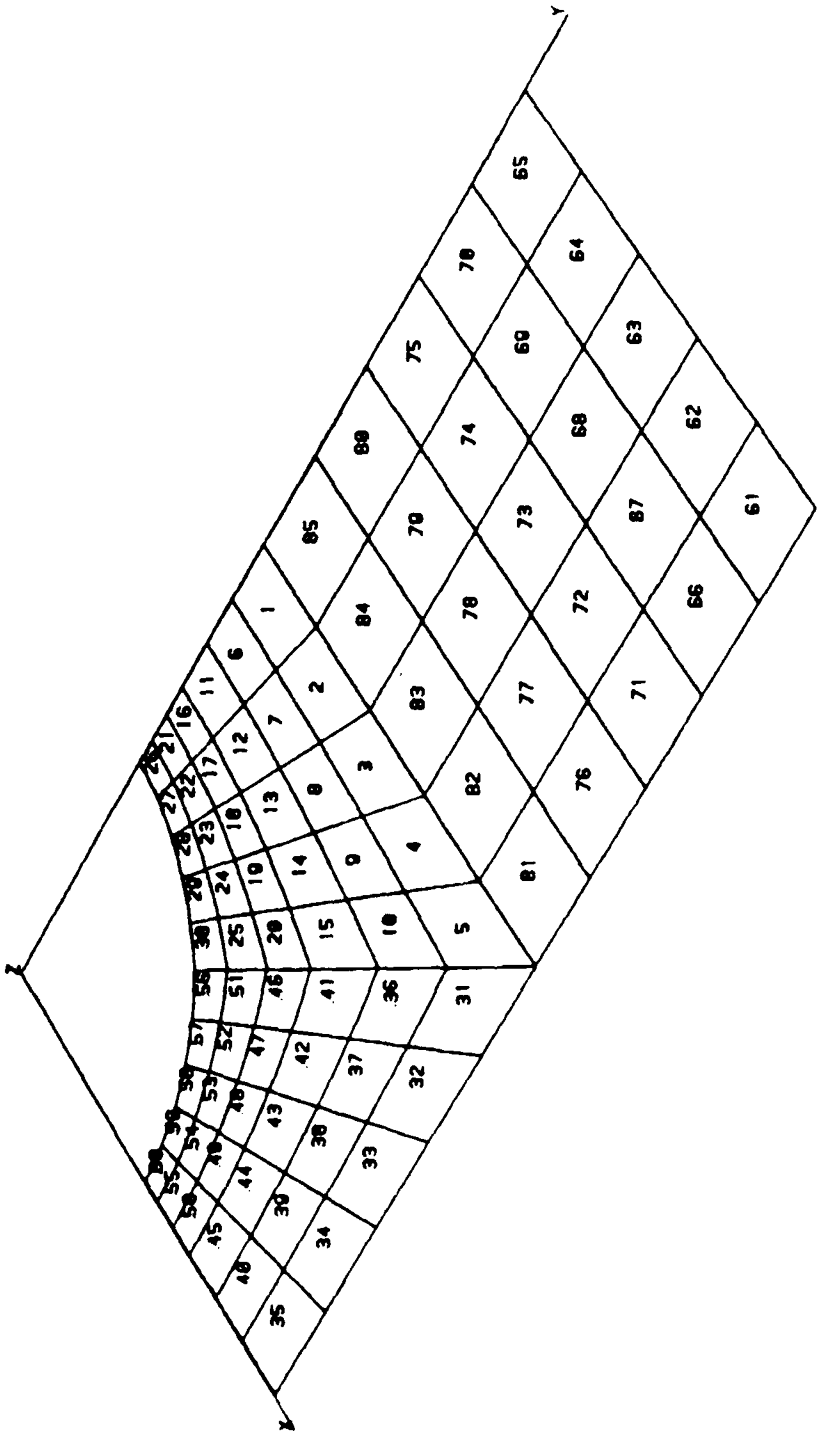
```
MAIN

 ::
? REFINE
 ENTER INPUT FILE #1 LABEL                    ::
? REFIN1
 ENTER OUTPUT FILE LABEL                      ::
? REFIN2
 CREATING NEW DATA FILE

   BEGIN REFINE ROUTINE...

 1 - RANDOM ELEMENT INPUT
 2 - RANGE OF ELEMENTS
 3 - EXECUTE REFINEMENT
 ENTER OPTION --------------->
? 2
 MAXIMUM NUMBER OF ELEMENTS
   SPECIFIABLE -------------->  150
 NUMBER STILL AVAILABLE ----->  150

 ENTER BEGINNING ELEMENT NUMBER, ENDING ELEMENT
   NUMBER, AND INCREMENT FOR RANGE.
 ENTER NBEG,NEND,INCR ------->
? 1,13,1

 1 - RANDOM ELEMENT INPUT
 2 - RANGE OF ELEMENTS
 3 - EXECUTE REFINEMENT
 ENTER OPTION --------------->
? 3

 ENTER CUT DIRECTION (R,S,T) >
? S

 MAXIMUM CUTS PER ELEMENT ---> 5
 ENTER CUTS PER ELEMENT ----->
? 4

 EQUALLY SPACED CUTS ? (Y,N) >
? Y

 REFINE COMPLETE.
 PERFORM SUBSEQUENT TIDY ? (Y,N) ->
? Y

 TIDY OPTION ONE OR TWO ? (OPT1, OPT2, HELP) ->
? OPT2

       TIDY COMPLETE
   110 DUPLICATE NODES DELETED

 ::
? PLOT
 ENTER INPUT FILE #1 LABEL                    ::
? REFIN2
 *
? ORIENT
   PLOT ORIENTATION AXES?(Y,N) .................:
? N
```

```
   *
? LABEL
    LABEL THE ELEMENTS?(Y,N) ......................:
? Y
    LABEL THE NODES?(Y,N) .........................:
? N
   *
? DRAW

    AT END OF PLOT, ENTER CHARACTER TO CONTINUE.
    READY (Y,N) ? ....:
? Y
```



```
 *
? MAIN

 ::
? LIST
 INVALID COMMAND, ENTER 'HELP' FOR LIST OF COMMANDS

 ::
? LIST

 LISTING OF DATA FILES AVAILABLE

 #10 --- REFIN1
 #11 --- PLAT2
 #12 --- REFIN2

 ::
? DELE
 ENTER LABEL                 ::
? REFIN1

 ::
? DELE
 ENTER LABEL                 ::
? PLAT2

 ::
? REFINE
 ENTER INPUT FILE #1 LABEL              ::
? REFIN2
 ENTER OUTPUT FILE LABEL                ::
? REFIN3
 CREATING NEW DATA FILE

   BEGIN REFINE ROUTINE...

 1 - RANDOM ELEMENT INPUT
 2 - RANGE OF ELEMENTS
 3 - EXECUTE REFINEMENT
 ENTER OPTION --------------->
? 2
 MAXIMUM NUMBER OF ELEMENTS
   SPECIFIABLE -------------->  150
 NUMBER STILL AVAILABLE ----->  150

 ENTER BEGINNING ELEMENT NUMBER, ENDING ELEMENT
   NUMBER, AND INCREMENT FOR RANGE.
 ENTER NBEG,NEND,INCR ------->
? 1,5,1

 1 - RANDOM ELEMENT INPUT
 2 - RANGE OF ELEMENTS
 3 - EXECUTE REFINEMENT
 ENTER OPTION --------------->
? 3

 ENTER CUT DIRECTION (R,S,T) >
? R

 MAXIMUM CUTS PER ELEMENT ---> 5
 ENTER CUTS PER ELEMENT ---- >
? 4
```

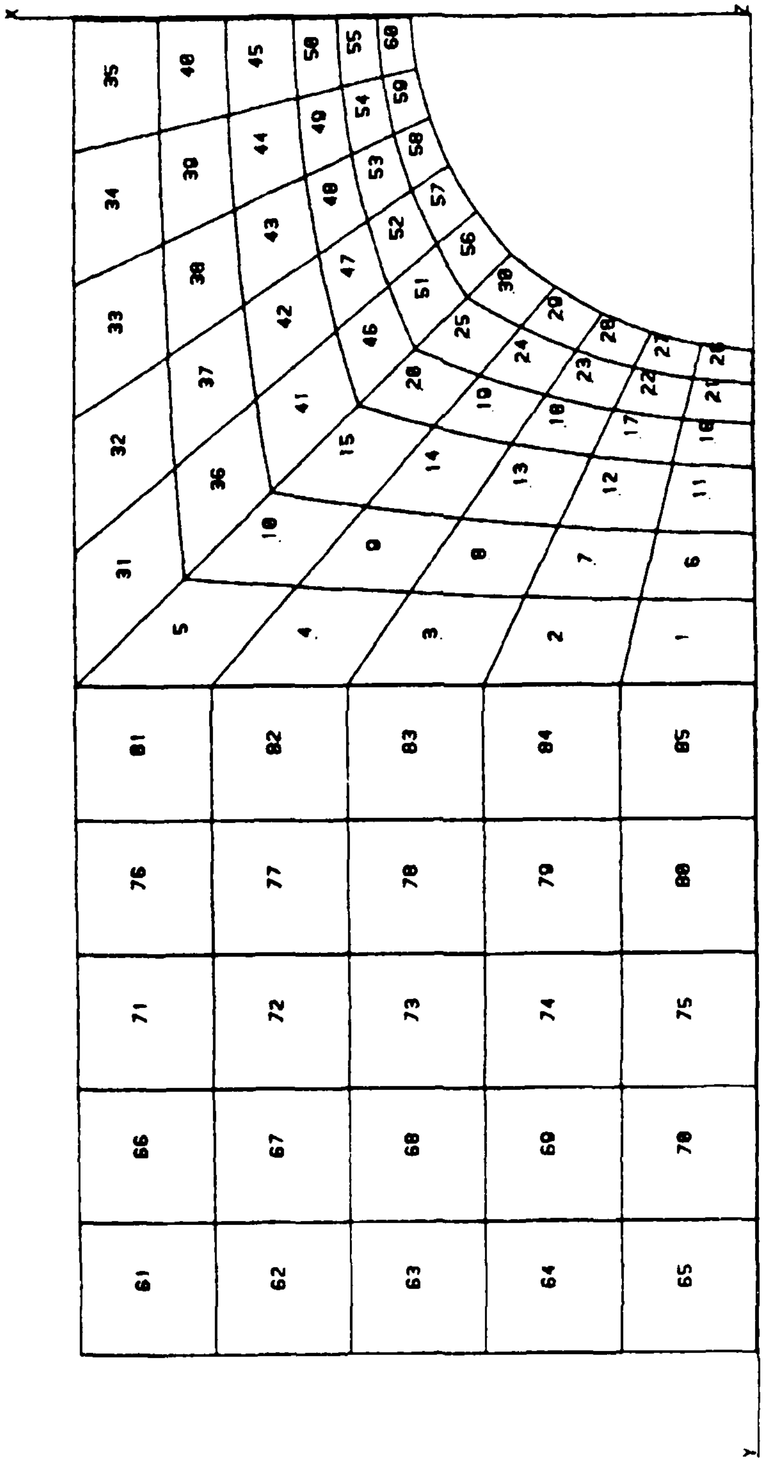
```
 EQUALLY SPACED CUTS ? (Y,N) >
? Y

 REFINE COMPLETE.
 PERFORM SUBSEQUENT TIDY ? (Y,N) ->
? Y

 TIDY OPTION ONE OR TWO ? (OPT1, OPT2, HELP) ->
? OPT2

       TIDY COMPLETE
    55 DUPLICATE NODES DELETED

 ::
? PLOT
 ENTER INPUT FILE #1 LABEL                ::
? REFIN3
  #
? DRAW

   AT END OF PLOT, ENTER CHARACTER TO CONTINUE.
   READY (Y,N) ? ....:
? Y
```



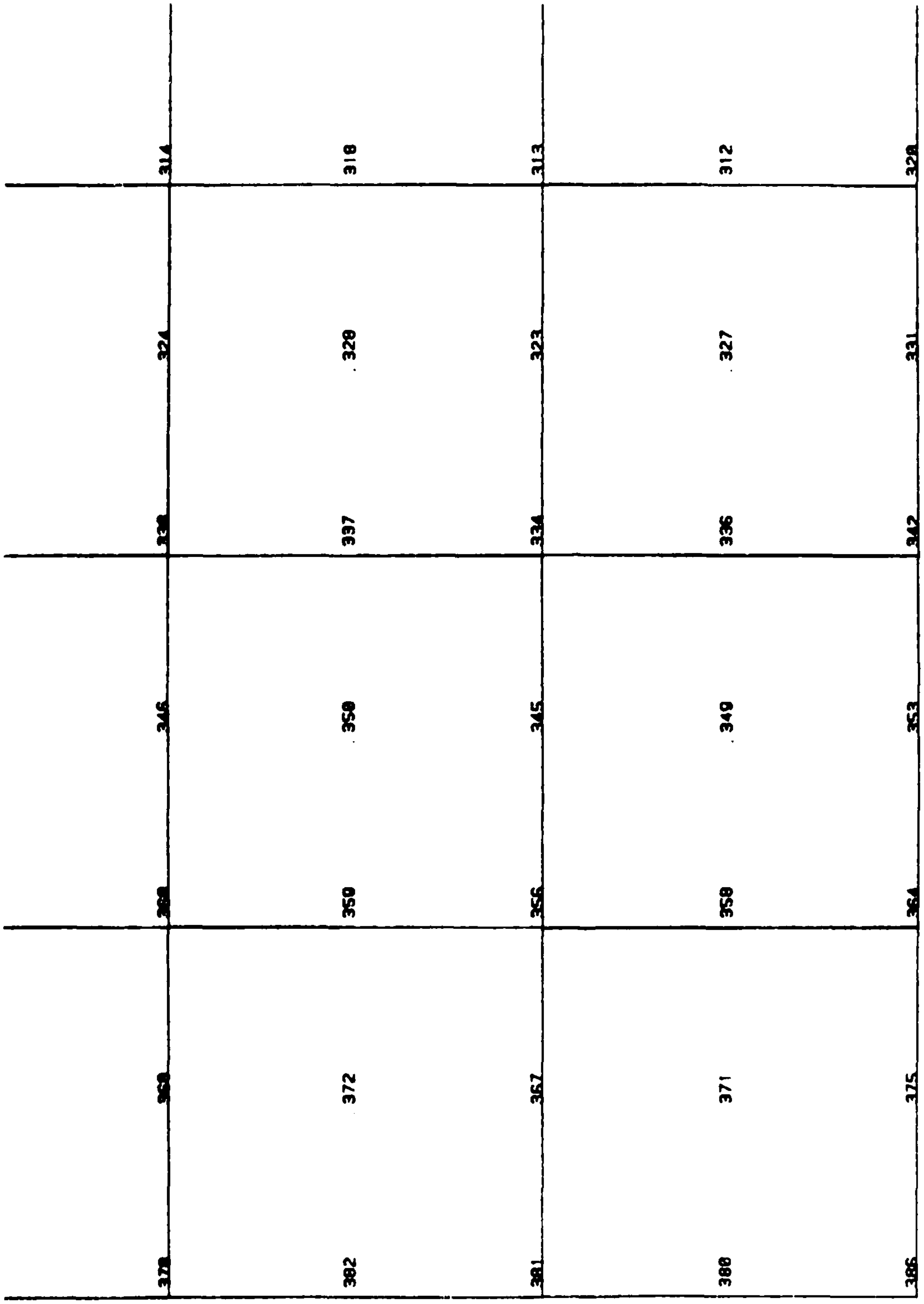
```
*
? MAIN

::
? LIST

 LISTING OF DATA FILES AVAILABLE

 #10 --- REFIN3
 #12 --- REFIN2

::
? DELE
 ENTER LABEL                    ::
? REFIN2

::
? RENUM
 ENTER INPUT FILE #1 LABEL                ::
? REFIN3
 ENTER OUTPUT FILE LABEL                  ::
? PLATE-4
 CREATING NEW DATA FILE

  NODE POINT REORDERING COMPLETE

  NODAL BANDWIDTH (OLD) =   97
  NODAL BANDWIDTH (NEW) =   97

::
? DELE
 ENTER LABEL                    ::
? REFIN3

::
? LIST

 LISTING OF DATA FILES AVAILABLE

 #11 --- PLATE-4

::
? BOUNDS
 ENTER INPUT FILE #1 LABEL                ::
? PLATE-4
 ENTER OUTPUT FILE LABEL                  ::
? PLATE+BCS
 CREATING NEW DATA FILE

 BOUNDARY CONDITIONS  MAY  BE ENTERED  FOR  RANDOM
   NODES,  RANGES OF NODES,  ALL NODES ON A PLANE,
   OR THE ENTIRE MODEL.

 FOR EACH GROUP OF NODES.  CONSTRAINTS  MAY BE SET
   IN ANY COMBINATION OF THE X, Y, & Z DIRECTIONS.
   (NOTE THAT 2-D MODELS MUST BE FIXED IN Z)

 CONSTRAINT DIRECTION CODE CONSTRUCTED AS FOLLOWS:

   ENTRY 1 - '1' FOR CONSTRAINED IN X DIRECTION
             '0' FOR UNCONSTRAINED IN X DIRECTION
   ENTRY 2 -     SAME FOR Y
   ENTRY 3 -     SAME FOR Z
```

```
  EXAMPLE -> '1,0,1' FOR CONSTRAINTS IN X AND Z
                     DIRECTIONS BUT NOT IN Y.

 1 - RANDOM NODES
 2 - RANGE OF NODES
 3 - SPECIFIED PLANE
 4 - ALL NODES
 5 - EXIT

 ENTER NODE SELECTION OPTION (1,2,...5) --------->
? 4

 ENTER CONSTRAINT DIRECTION CODE (THREE VALUES) ->
? 0 0 0

 1 - RANDOM NODES
 2 - RANGE OF NODES
 3 - SPECIFIED PLANE
 4 - ALL NODES
 5 - EXIT

 ENTER NODE SELECTION OPTION (1,2,...5) --------->
? 3

 ENTER CONSTRAINT DIRECTION CODE (THREE VALUES) ->
? 1 0 0

 PLANE IS DEFINED BY  AX + BY + CZ = D.
   ENTER COEFFICIENTS ( A , B , C , D) ---------->
? 1 0 0 0

             24 NODES FOUND

 1 - RANDOM NODES
 2 - RANGE OF NODES
 3 - SPECIFIED PLANE
 4 - ALL NODES
 5 - EXIT

 ENTER NODE SELECTION OPTION (1,2,...5) --------->
? 3

 ENTER CONSTRAINT DIRECTION CODE (THREE VALUES) ->
? 0 1 0

 PLANE IS DEFINED BY  AX + BY + CZ = D.
   ENTER COEFFICIENTS ( A , B , C , D) ---------->
? 0 1 0 0

             14 NODES FOUND

 1 - RANDOM NODES
 2 - RANGE OF NODES
 3 - SPECIFIED PLANE
 4 - ALL NODES
 5 - EXIT

 ENTER NODE SELECTION OPTION (1,2,...5) --------->
? 5
```

```
            30 BOUNDARY CONDITIONS ADDED
             30 BOUNDARY CONDITIONS TOTAL

  ::
? LIST

 LISTING OF DATA FILES AVAILABLE

 #10 --- PLATE+BCS
 #11 --- PLATE-4

  ::
? DELE
 ENTER LABEL                ::
? PLATE-4

  ::
? PLOT
 ENTER INPUT FILE #1 LABEL              ::
? PLATE+BCS
  *
? VERT
    WHICH AXIS IS VERTICAL?
    ENTER 1 FOR X, 2 FOR Y, OR 3 FOR Z ..........:
? 1
  *
? EYE
    ENTER THE EYE POSITION,
    XEYE, YEYE, ZEYE ............................:
? 0 0 10000
  *
? DRAW

   AT END OF PLOT, ENTER CHARACTER TO CONTINUE.
   READY (Y,N) ? ....:
? Y
```



```
  $
? ZOOM
    DO YOU WANT THE ZOOM FUNCTION (Y,N) ? ......:
? Y
    SCALE THE ZOOM AREA (Y,N) ? ................:
? Y
    DIGITIZE THE LOWER LEFT CORNER
    AND THE UPPER RIGHT CORNER OF THE
    ZOOM AREA
    READY?(Y,N) ................................:
? Y
? LABEL
    LABEL THE ELEMENTS?(Y,N) ....................:
? Y
    LABEL THE NODES?(Y,N) .......................:
? Y
    LABEL ALL THE SURFACES?(Y,N) ................:
? Y
  $
? AXES
    PLOT AND LABEL THE AXES?(Y,N) ...............:
? N
  $
? DRAW

   AT END OF PLOT, ENTER CHARACTER TO CONTINUE.
   READY (Y,N) ? ....:
? Y
```



```
 *
? ZOOM
   DO YOU WANT THE ZOOM FUNCTION (Y,N) ? ......!
? N
 *
? MAIN

 ::
? LIST

 LISTING OF DATA FILES AVAILABLE

 #18 --- PLATE+BCS

 ::
? LOAD
 ENTER INPUT FILE #1 LABEL                 ::
? PLATE+BCS
 ENTER OUTPUT FILE LABEL                   ::
? FINAL
 CREATING NEW DATA FILE

 LOAD SPECIFICATION OPTIONS
   N - NODAL LOAD ENTRY
   E - ELEMENT LOAD ENTRY
   L - LIST EXISTING LOADS
   H - PRINTS THIS LIST
   S - STOP LOAD ENTRY

 ENTER LOAD SPECIFICATION OPTION     --->
? N

 NODE SPECIFICATION OPTIONS
   S - SINGLE NODE
   R - RANGE OF NODES
   P - ALL NODES ON A GIVEN PLANE
   H - PRINTS THIS LIST
   E - EXIT NODAL LOAD SPECIFICATION SECTION

 ENTER NODE SPECIFICATION OPTION     --->
? S
 ENTER CASE NUMBER                   --->
? 1
 ENTER FORCE VECTOR (FX,FY,FZ)       --->
? 0 100 0
 NODE NUMBER TO BE LOADED ?          --->
? 306

 ENTER NODE SPECIFICATION OPTION     --->
? E

 LEAVING NODAL LOAD SPECIFICATION SECTION

 ENTER LOAD SPECIFICATION OPTION     --->
? H
```

```
LOAD SPECIFICATION OPTIONS
  N - NODAL LOAD ENTRY
  E - ELEMENT LOAD ENTRY
  L - LIST EXISTING LOADS
  H - PRINTS THIS LIST
  S - STOP LOAD ENTRY

ENTER LOAD SPECIFICATION OPTION    --->
? S

LEAVING LOAD SPECIFICATION MODULE

 ::
? LIST

LISTING OF DATA FILES AVAILABLE

010 --- PLATE+BCS
011 --- FINAL

 ::
? DELE
ENTER LABEL                    ::
? PLATE+BCS

 ::
? STOP

LISTING OF DATA FILES AVAILABLE

011 --- FINAL
```

```
PRE-PROCESSOR UTILITIES

 1. AGRID
 2. CORGEN
 3. CREATE
 4. EXPAND
 5. GPLOT
 6. HIDLIN
 7. IJKGEN
 8. NEUTRAL
 9. PREP
10. REFMT
11. SPATCH
12. SURFDIG
13. TRNSFR
14. FILE UTILITIES

SELECT BY NUMBER OR TYPE Q TO QUIT ? 14

FILE PROCESSING UTILITIES

 1. ENQUIRE,F
 2. INPUT / OUTPUT FILE NAMES
 3. RENAME
 4. RETURN TO MAIN MENU

SELECT BY NUMBER OR TYPE Q TO QUIT ? 3
ENTER PRESENT FILE NAME? TAPE11
ENTER DESIRED FILE NAME? UNFMT

FILE PROCESSING UTILITIES

 1. ENQUIRE,F
 2. INPUT / OUTPUT FILE NAMES
 3. RENAME
 4. RETURN TO MAIN MENU

SELECT BY NUMBER OR TYPE Q TO QUIT ? 4

PRE-PROCESSOR UTILITIES

 1. AGRID
 2. CORGEN
 3. CREATE
 4. EXPAND
 5. GPLOT
 6. HIDLIN
 7. IJKGEN
 8. NEUTRAL
 9. PREP
10. REFMT
11. SPATCH
12. SURFDIG
13. TRNSFR
14. FILE UTILITIES

SELECT BY NUMBER OR TYPE Q TO QUIT ? 10
```

```
*****************************************

BEGIN REFMT   --   MAGNA INPUT GENERATOR

*****************************************


** SUMMARY OF INITIAL SCAN OF DATA FILE **

-----------------------------------------

NUMBER OF NODAL POINTS .......   380
NUMBER OF ELEMENTS (TOTAL) ...    85
NUMBER OF CONSTRAINT RECORDS..    39
NUMBER OF LINEAR CONSTRAINTS..     0
NUMBER OF NODAL LOADS ........     1
NUMBER OF ELEMENT LOADS ......     0
DISTINCT LOAD CASES / GROUPS..     1


    MAGNA        NUMBER OF    ELEMENTS WITH
ELEMENT TYPE      ELEMENTS    UNSPEC. MATRL

      9              85             85


ENTER A THREE-LINE PROBLEM TITLE   (UP TO 80 CHARACTERS PER LINE)

.........1.........2.........3.........4.........5.........6.........7.........8

?  THIN PERFORATED PLATE - PLANE STRESS - LINEAR, STATIC ANALYSIS
?  LENGTH = 20., WIDTH = 10., FOR QUADRANT MODELED, HOLE RADIUS = 5.0
?  9-NODE PLANE STRESS ELEMENTS, 85 ELEMENTS TOTAL

*********************************************

 MAJOR SOLUTION OPTIONS AND PARAMETERS

*********************************************

 ENTER ANALYSIS TYPE      (LINEAR,NONLINEAR) -
? LINEAR

 ENTER ANALYSIS SUBTYPE  (STATIC,DYNAMIC)----
? STATIC

 INCLUDE THERMAL EFFECTS (Y/N) --------------
? N

 POSTPROCESSOR FILE TO BE WRITTEN (Y/N) -----
? Y
```

```
*********************************************
     END OF OPTIONS SPECIFICATIONS
*********************************************


INDIVIDUAL ELEMENTS IN THE MODEL CONTAINUNDEFINED PROPERTIES

ELEMENT TYPE       =     9
NUMBER OF ELEMENTS=     86

PLEASE DEFINE A DEFAULT PROPERTY CODE FOR THIS ELEMENT TYPE,
OR ENTER MATERIALS DATA DIRECTLY BELOW.


MATERIAL PROPERTY DEFINITION OPTIONS

 E -- ENTER PROPERTY DATA DIRECTLY
 C -- SPECIFY A LIBRARY PROPERTY CODE
 L -- LIST SELECTED LIBRARY ENTRIES

ENTER OPTION  ( E , C , L ) ------------
? L

LIBRARY MATERIAL DESCRIPTIONS CAN BE LISTED BY MATERIAL TYPE
VALID MATERIAL TYPES ARE AS FOLLOWS ----

ACRYL - ACRYLICS
ALUMI - ALUMINUM ALLOYS
CASTI - CAST IRONS
COPPR - COPPER-BASED ALLOYS
GLASS - GLASSES
MAGNS - MAGNESIUM ALLOYS
NICKL - NICKEL ALLOYS
PLYMR - POLYMERIC MATERIALS
POLYC - POLYCARBONATES
STEEL - CARBON STEELS
STSTL - STAINLESS STEELS
TITNM - TITANIUM

ENTER MATERIAL TYPE (STEEL,STSTL,ETC.)--
? ALUMI
```

```
MATL. CODE  ...........DESCRIPTION................

   00200    ALUMI - UNS A0 2024-0
   00202    ALUMI - UNS A0 2024-T36
   00205    ALUMI - UNS A0 2024-T62  SHEET
   00210    ALUMI - UNS A0 2021-T81  BAR
   00211    ALUMI - UNS A0 2021-T81  SHEET
   00220    ALUMI - UNS A0 5052-O
   00223    ALUMI - UNS A0 5052-H32
   00230    ALUMI - UNS A0 6061-O
   00231    ALUMI - UNS A0 6061-T4
   00232    ALUMI - UNS A0 6061-T6
   00240    ALUMI - UNS A0 7075-T6

MATERIAL PROPERTY DEFINITION OPTIONS

 E -- ENTER PROPERTY DATA DIRECTLY
 C -- SPECIFY A LIBRARY PROPERTY CODE
 L -- LIST SELECTED LIBRARY ENTRIES

ENTER OPTION  ( E , C , L ) ------------
? C

ENTER LIBRARY PROPERTY CODE --
? 205

MATERIAL PROPERTIES DEFINITION FOR THE MODEL IS COMPLETE.
AT THIS POINT MATERIALS DATA MAY BE EDITED AS NECESSARY.
(NOTE THAT SOME DATA WHICH IS UNIMPORTANT FOR THE CURRENT
ANALYSIS MAY BE DEFINED AS ZERO)

CURRENT PROPERTIES ARE LISTED BELOW ----


CODE  MODULUS   POIS.RATIO  DENSITY   YIELD STR. THERM.EXP.

 -9  .1050E+08  .3300E+00  .2500E-03  .4900E+05  .1260E-04

L = (L)IST CURRENT PROPERTIES TABLE
C = (C)HANGE AN ENTRY IN THE TABLE
S = (S)TOP EDITING

ENTER OPTION  ( L , C , S ) ------------
? S
```

```
***************************************
*                                     *
*      DATA GENERATION COMPLETE       *
*                                     *
***************************************


PRE-PROCESSOR UTILITIES

 1. AGRID
 2. CORGEN
 3. CREATE
 4. EXPAND
 5. GPLOT
 6. HIDLIN
 7. IJKGEN
 8. NEUTRAL
 9. PREP
10. REFMT
11. SPATCH
12. SURFDIG
13. TRNSFR
14. FILE UTILITIES

 SELECT BY NUMBER OR TYPE Q TO QUIT ? Q
/ENQUIRE,F

 LOCAL FILE INFORMATION.

  FILENAME  LENGTH/PRUS  TYPE  STATUS    FS

   INPUT*          1     IN.*  EOR
   INPUT                 LO.
   OUTPUT                LO.
   UTILS          59     LO.   EOR
   UNFMT          83     LO.   EOR
   FDATA          46     LO.   BOI

  TOTAL = 6
/RETURN,UNFMT
RETURN,UNFMT.
/SAVE,FDATA=SHEET
/
```

REVISED OUTPUT LISTINGS FOR SECTION 7.3

PART II

```
GET,UTILS/UN=0820130
/UTILS

 PROBLEMS WITH MISSING PROGRAM FILE (FILE NOT FOUND)
 SHOULD BE REFFERED TO  TSGT. S. ZASTROW, 255-6100.


 MAGNA PRE/POST PROCESSOR UTILITIES

  1. PRE-PROCESSORS
  2. POST-PROCESSORS

 SELECT BY NUMBER OR TYPE Q TO QUIT ? 1

 PRE-PROCESSOR UTILITIES

  1. AGRID
  2. CORGEN
  3. CREATE
  4. EXPAND
  5. GPLOT
  6. HIDLIN
  7. IJKGEN
  8. NEUTRAL
  9. PREP
 10. REFMT
 11. SPATCH
 12. SURFDIG
 13. TRNSFR
 14. FILE UTILITIES

 SELECT BY NUMBER OR TYPE Q TO QUIT ? 7


          ****************************************
          **********    BEGIN IJKGEN    **********
          ****************************************

 I J K G E N   -   GENERATION OF GEOMETRIC MESH DATA FOR SOLID,
 THICK SHELL OR PLATE FINITE ELEMENT MODELS, USING AN INTEGER -
 COORDINATE INDEXING SCHEME.  OPTIONAL USER ROUTINES ARE -
   (1)  SURFAC (I,J,K,ALPHA,BETA,ZETA) - DEFINE MESH GEOMETRY
   (2)  CRDTRN (ALPHA,BETA,ZETA,X,Y,Z) - COORD. TRANSFORMATION
 BUILT-IN OPTIONS INCLUDE RECTANGULAR, CYLINDRICAL OR SPHERICAL
 COORDINATES  AND UNIFORM OR PROPORTIONALLY GRADED MESH SPACING

   (3)  UINPUT - USER PARAMETER INPUT ROUTINE (INITIALIZE DATA
                 IN BLANK COMMON)

 ..............................................................

 **********  USER SUBROUTINE 'SURFAC' NOT GIVEN  **********

 BUILT-IN MESH DIVISION OPTIONS ARE AS FOLLOWS -
  (1) - UNIFORM MESH IN EACH DIRECTION
  (2) - GRADED MESH (SPECIFY RATIO OF FIRST/LAST ELEMENT SIZE

 ENTER OPTION ( 1 , 2 ) ................
? 2
```

```
ENTER THE RATIO OF FIRST / LAST ELEMENT LENGTHS FOR EACH
COORDINATE DIRECTION (ALPHA, BETA, ZETA) (R=1 FOR UNIFORM)

ENTER LENGTH RATIOS (R1,R2,R3) ........
? 4., 1., 1.0

**********  USER SUBROUTINE 'CRDTRN' NOT GIVEN  **********

BUILT-IN COORDINATE SYSTEM TRANSFORMATION OPTIONS ARE - - -
(1) RECTANGULAR,  (2) CYLINDRICAL,  (3) SPHERICAL

ENTER COORDINATE SYSTEM OPTION (1,2,3) -
? 1

** PLEASE NOTE THE FOLLOWING CONVENTIONS FOR RECTANGULAR SYSTEM **

 'ALPHA' = X      'BETA' = Y      'ZETA' = Z
A RIGHT-HANDED SYSTEM IS ASSUMED.

ENTER LIMITING SURFACE COORDINATE VALUES -

 1. ALPHA(MIN)    2. ALPHA(MAX)
 3. BETA (MIN)    4. BETA (MAX)
 5. ZETA (MIN)    6. ZETA (MAX)

? 0.0, 20.0, 0.0, 16.0, 0.0, 0.2

ENTER THE NUMBER OF ELEMENTS TO BE GEN-
 ERATED IN THE ALPHA, BETA AND ZETA CO-
 ORDINATE DIRECTIONS, RESPECTIVELY ....
? 6, 5, 1

*** BEGIN GENERATION PHASE ***

NUMBER OF NODES TO BE GENERATED    =  420
NUMBER OF ELEMENTS TO BE GENERATED=   30

THE THICKNESS DIRECTION OF THE MODEL (IF ONE EXISTS, AS IN A
THICK SHELL) MUST BE IDENTIFIED TO ORIENT ELEMENTS PROPERLY.

OPTIONS ARE (1)ALPHA, (2)BETA, (3)ZETA, OR (4)<UNIMPORTANT>.

ENTER THICKNESS DIRECTION CODE (1,2,3,4) -
? 3
```

```
*** DATA GENERATION COMPLETE ***

****** IJKGEN  TERMINATED ******

PRE-PROCESSOR UTILITIES

 1. AGRID
 2. CORGEN
 3. CREATE
 4. EXPAND
 5. GPLOT
 6. HIDLIN
 7. IJKGEN
 8. NEUTRAL
 9. PREP
10. REFMT
11. SPATCH
12. SURFDIG
13. TRNSFR
14. FILE UTILITIES

SELECT BY NUMBER OR TYPE Q TO QUIT ? 14

FILE PROCESSING UTILITIES

 1. ENQUIRE,F
 2. INPUT / OUTPUT FILE NAMES
 3. RENAME
 4. RETURN TO MAIN MENU

SELECT BY NUMBER OR TYPE Q TO QUIT ? 3
ENTER PRESENT FILE NAME? UNFMT
ENTER DESIRED FILE NAME? TAPE11

FILE PROCESSING UTILITIES

 1. ENQUIRE,F
 2. INPUT / OUTPUT FILE NAMES
 3. RENAME
 4. RETURN TO MAIN MENU

SELECT BY NUMBER OR TYPE Q TO QUIT ? 4

PRE-PROCESSOR UTILITIES

 1. AGRID
 2. CORGEN
 3. CREATE
 4. EXPAND
 5. GPLOT
 6. HIDLIN
 7. IJKGEN
 8. NEUTRAL
 9. PREP
10. REFMT
11. SPATCH
12. SURFDIG
13. TRNSFR
14. FILE UTILITIES

SELECT BY NUMBER OR TYPE Q TO QUIT ? 7
```

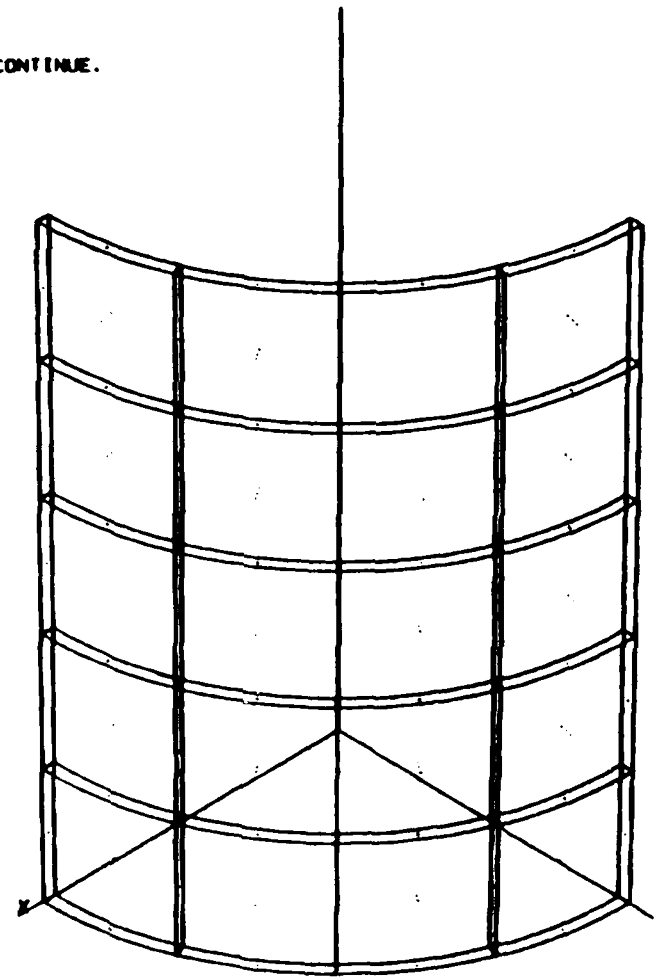
```
          ****************************************
          **********    BEGIN IJKGEN    **********
          ****************************************

I J K G E N   -   GENERATION OF GEOMETRIC MESH DATA FOR SOLID,
THICK SHELL OR PLATE FINITE ELEMENT MODELS, USING AN INTEGER -
COORDINATE INDEXING SCHEME.  OPTIONAL USER ROUTINES ARE -
  (1)  SURFAC (I,J,K,ALPHA,BETA,ZETA) - DEFINE MESH GEOMETRY
  (2)  CRDTRN (ALPHA,BETA,ZETA,X,Y,Z) - COORD. TRANSFORMATION
BUILT-IN OPTIONS INCLUDE RECTANGULAR, CYLINDRICAL OR SPHERICAL
COORDINATES, AND UNIFORM OR PROPORTIONALLY GRADED MESH SPACING

  (3)  UINPUT - USER PARAMETER INPUT ROUTINE (INITIALIZE DATA
              IN BLANK COMMON)

.............................................................

**********  USER SUBROUTINE 'SURFAC' NOT GIVEN  **********

BUILT-IN MESH DIVISION OPTIONS ARE AS FOLLOWS -
 (1) - UNIFORM MESH IN EACH DIRECTION
 (2) - GRADED MESH (SPECIFY RATIO OF FIRST/LAST ELEMENT SIZE

ENTER OPTION ( 1 , 2 ) ................
? 1

**********  USER SUBROUTINE 'CRDTRN' NOT GIVEN  **********

BUILT-IN COORDINATE SYSTEM TRANSFORMATION OPTIONS ARE - - -
(1) RECTANGULAR,  (2) CYLINDRICAL,  (3) SPHERICAL

ENTER COORDINATE SYSTEM OPTION (1,2,3) -
? 2

** PLEASE NOTE THE FOLLOWING CONVENTIONS FOR CYLINDRICAL SYSTEM **

 'ALPHA' = RADIUS     'BETA' = ANGLE     'ZETA' = AXIAL
SYSTEM IS RIGHT-HANDED, WITH ALL ANGLES MEASURED IN DEGREES.

ENTER LIMITING SURFACE COORDINATE VALUES -

 1. ALPHA(MIN)    2. ALPHA(MAX)
 3. BETA (MIN)    4. BETA (MAX)
 5. ZETA (MIN)    6. ZETA (MAX)

? 5.0, 5.2, 0.0, 90.0, 0.0, 10.0

ENTER THE NUMBER OF ELEMENTS TO BE GEN-
 ERATED IN THE ALPHA, BETA AND ZETA CO-
 ORDINATE DIRECTIONS, RESPECTIVELY ....
? 1, 4, 5
```

```
*** BEGIN GENERATION PHASE ***

NUMBER OF NODES TO BE GENERATED    =  207
NUMBER OF ELEMENTS TO BE GENERATED=   20

THE THICKNESS DIRECTION OF THE MODEL (IF ONE EXISTS, AS IN A
THICK SHELL) MUST BE IDENTIFIED TO ORIENT ELEMENTS PROPERLY.

OPTIONS ARE (1)ALPHA, (2)BETA, (3)ZETA, OR (4)<UNIMPORTANT>.

ENTER THICKNESS DIRECTION CODE (1,2,3,4) -
? 1

*** DATA GENERATION COMPLETE ***

****** IJKGEN  TERMINATED ******

PRE-PROCESSOR UTILITIES

 1. AGRID
 2. CORGEN
 3. CREATE
 4. EXPAND
 5. GPLOT
 6. HIDLIN
 7. IJKGEN
 8. NEUTRAL
 9. PREP
10. REFMT
11. SPATCH
12. SURFDIG
13. TRNSFR
14. FILE UTILITIES

SELECT BY NUMBER OR TYPE Q TO QUIT ? 14

FILE PROCESSING UTILITIES

 1. ENQUIRE.F
 2. INPUT / OUTPUT FILE NAMES
 3. RENAME
 4. RETURN TO MAIN MENU

SELECT BY NUMBER OR TYPE Q TO QUIT ? 3
ENTER PRESENT FILE NAME? UNFMT
ENTER DESIRED FILE NAME? TAPE10

FILE PROCESSING UTILITIES

 1. ENQUIRE.F
 2. INPUT / OUTPUT FILE NAMES
 3. RENAME
 4. RETURN TO MAIN MENU

SELECT BY NUMBER OR TYPE Q TO QUIT ? 4
```

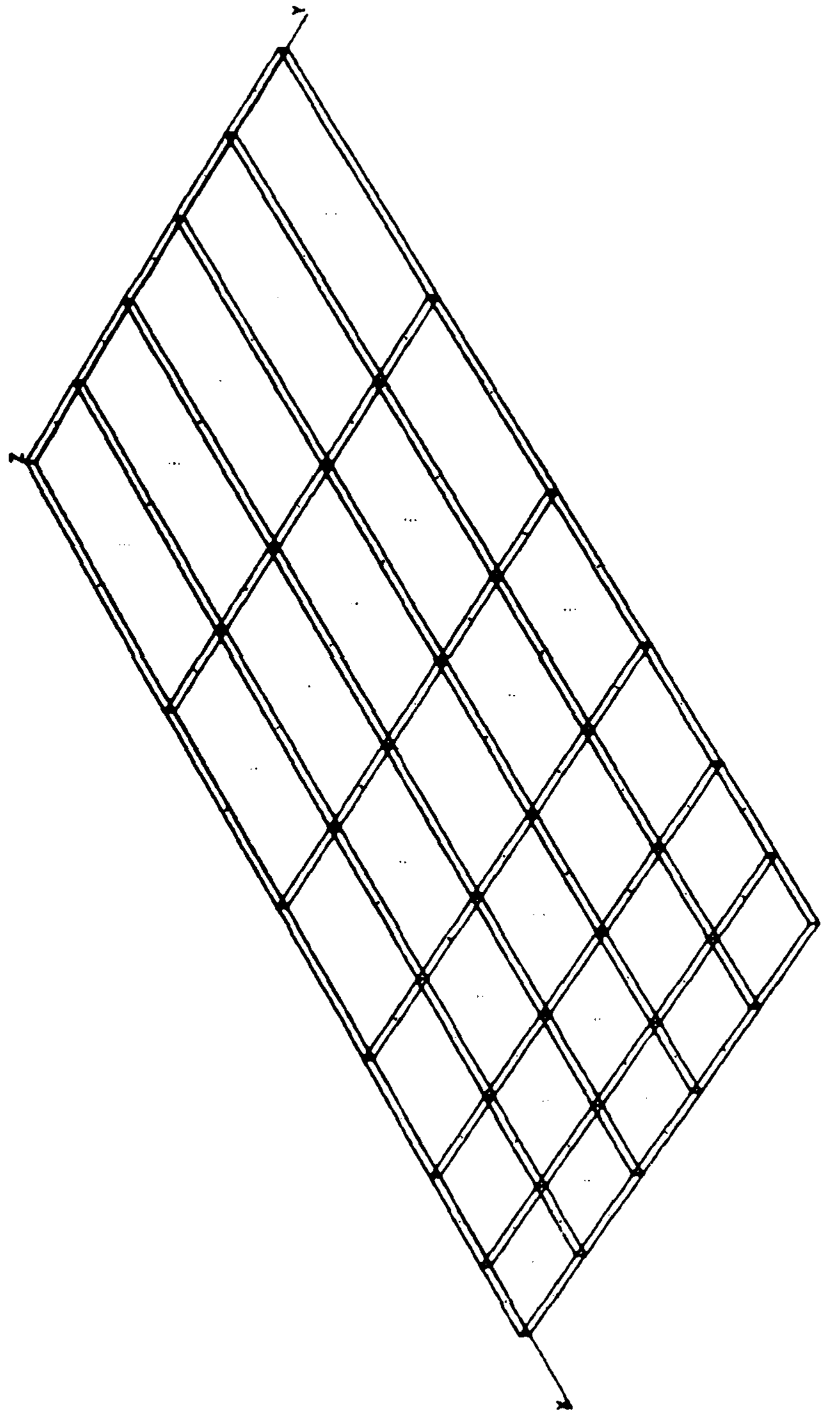
```
PRE-PROCESSOR UTILITIES

  1. AGRID
  2. CORGEN
  3. CREATE
  4. EXPAND
  5. GPLOT
  6. HIDLIN
  7. IJKGEN
  8. NEUTRAL
  9. PREP
 10. REFMT
 11. SPATCH
 12. SURFDIG
 13. TRNSFR
 14. FILE UTILITIES

SELECT BY NUMBER OR TYPE 0 TO QUIT ? 9

          ******************************************
          *************  BEGIN PREP  ***************
          **** MAGNA PREPROCESSING UTILITIES *******
          ******************************************


 PLOTTING TERMINAL ( Y , N ) ?::
? Y
 TEKTRONIX OR HP ( T , H ) ?  ::
? T

 TEKTRONIX TERMINAL TYPES ----
  0. 4006-1
  1. 4010 / 4012 / 4013 / 4052
  2. 4014 / 4015
  3. 4014 / 4015   ( ENH. GR. MOD. )
  4. 4114

 ENTER TERMINAL TYPE ( 0 - 4 )::
? 4
 ENTER CHARACTERS PER SECOND  ::
? 120
 NUMBER OF INPUT DATA FILES ? ::
? 2
 ENTER FILE #                 ::
? 10
 ENTER LABEL                  ::
? CYL-1
 ENTER FILE #                 ::
? 11
 ENTER LABEL                  ::
? PLT-1
```

```
 TYPE 'HELP' FOR LIST OF COMMANDS

 ::
? LIST

 LISTING OF DATA FILES AVAILABLE

 010 --- CYL-1
 011 --- PLT-1

 ::
? PLOT
 ENTER INPUT FILE #1 LABEL              ::
? CYL-1

 INITIALIZATION OF PLOTTING ROUTINE ERASES SCREEN.
 READY (Y,N) ? ....:
? Y
```

```
  &
? AXES
   PLOT AND LABEL THE AXES?(Y,N) ...............!
? Y
  &
? DRAW

  AT END OF PLOT. ENTER CHARACTER TO CONTINUE.
  READY (Y,N) ? ....!
? Y
```

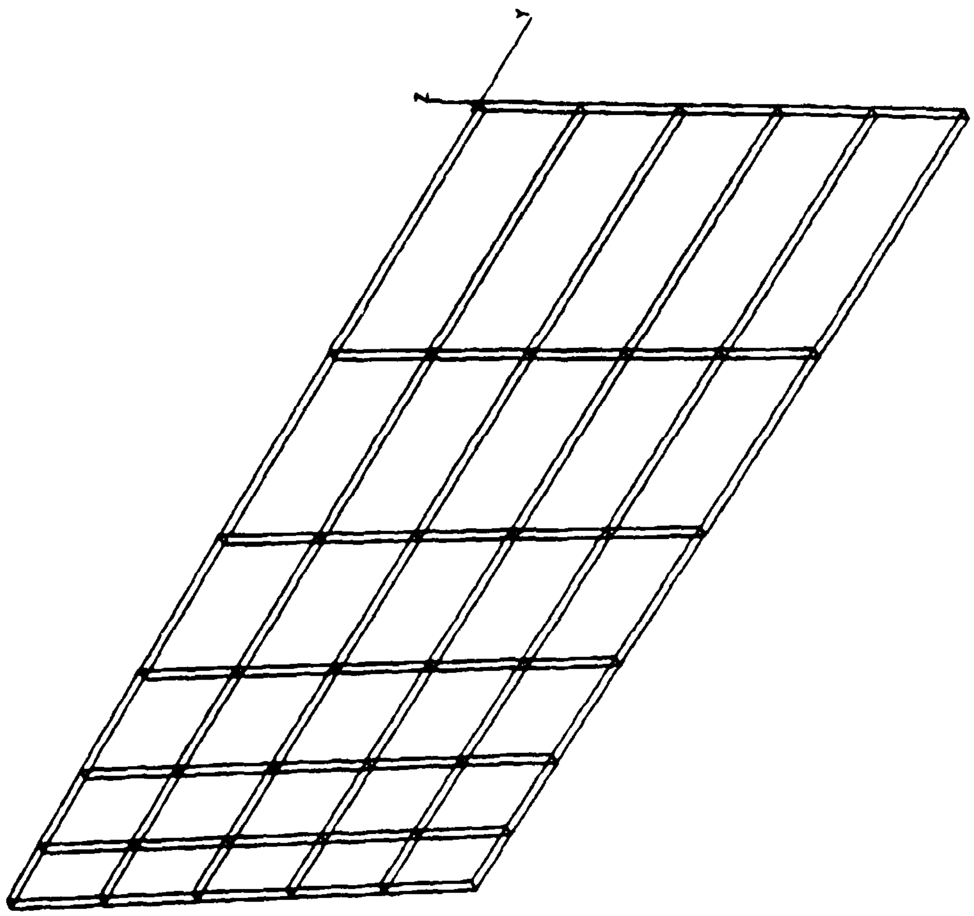


```
     *
 ? MAIN

  ::
 ? MASK
  ENTER INPUT FILE #1 LABEL                ::
 ? CYL-1
  ENTER OUTPUT FILE LABEL                  ::
 ? CYL-1GN
  CREATING NEW DATA FILE
  MASK OPTIONS -
    1. 3-D ELEMENTS ONLY
    2. 2-D ELEMENTS ONLY
  ENTER OPTION...::
 ? 1
  MAX NODE POSITION # IN FINAL ELEMENTS?
    (8,16,20,26,27) ::
 ? 16

  MASK COMPLETE.
  PERFORM SUBSEQUENT SIFT ? (Y,N) ->
 ? Y

        SIFT COMPLETE
    130 UNUSED NODES DELETED

  ::
 ? PLOT
  ENTER INPUT FILE #1 LABEL                ::
 ? PLT-1
     *
 ? DRAW

    AT END OF PLOT, ENTER CHARACTER TO CONTINUE.
    READY (Y,N) ? ....:
 ? Y
```

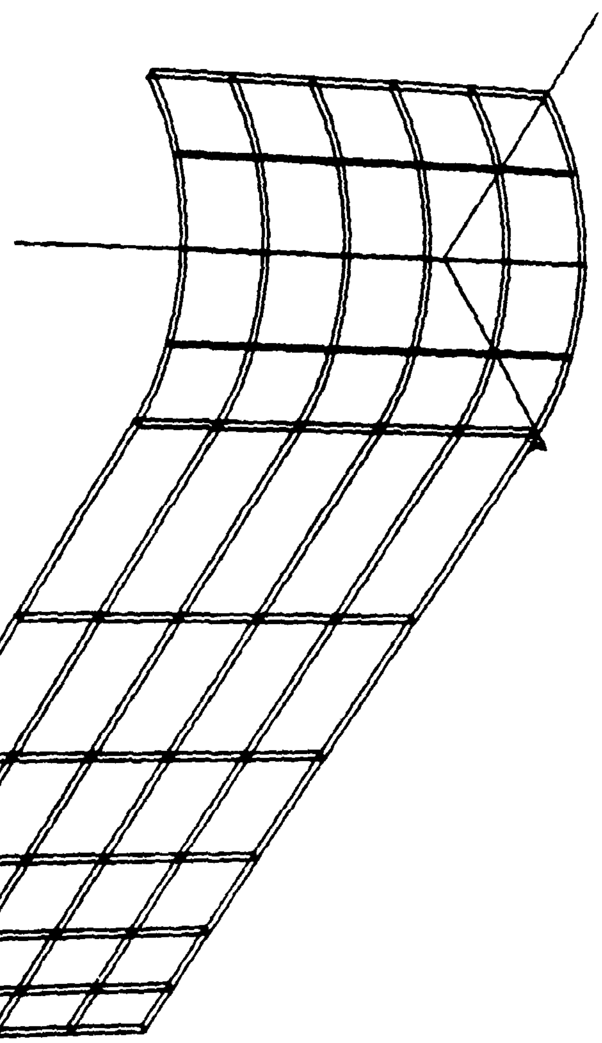


```
 &
? MAIN

 ::
? MASK
 ENTER INPUT FILE #1 LABEL                ::
? PLT-1
 ENTER OUTPUT FILE LABEL                  ::
? PLT-16N
 CREATING NEW DATA FILE
 MASK OPTIONS -
   1. 3-D ELEMENTS ONLY
   2. 2-D ELEMENTS ONLY
 ENTER OPTION...::
? 1
 MAX NODE POSITION # IN FINAL ELEMENTS?
   (8,16,20,26,27) ::
? 16

 MASK COMPLETE.
 PERFORM SUBSEQUENT SIFT ? (Y,N) ->
? Y

      SIFT COMPLETE
   203 UNUSED NODES DELETED

 ::
? LIST

 LISTING OF DATA FILES AVAILABLE

 #10 --- CYL-1
 #11 --- PLT-1
 #12 --- CYL-16N
 #13 --- PLT-16N

 ::
? DELETE
 ENTER LABEL                ::
? CYL-1

 ::
? DELETE
 ENTER LABEL                ::
? PLT-1

 ::
? ROTATE
 ENTER INPUT FILE #1 LABEL                ::
? PLT-16N
 ENTER OUTPUT FILE LABEL                  ::
? PLT-ROT
 CREATING NEW DATA FILE
 ROTATIONS PERFORMED ABOUT X, THEN Y, THEN Z.
 ENTER ROTATION ANGLES IN DEGREES (X,Y,Z) -->
? -90., 0.0, -90.
```

```
 !!
? LIST

 LISTING OF DATA FILES AVAILABLE

 010 --- PLT-ROT
 012 --- CYL-10N
 013 --- PLT-10N

 !!
? PLOT
 INVALID COMMAND, ENTER 'HELP' FOR LIST OF COMMANDS

 !!
? PLOT
 ENTER INPUT FILE #1 LABEL                 !!
? PLT-ROT
  0
? DRAW

   AT END OF PLOT, ENTER CHARACTER TO CONTINUE.
   READY (Y,N) ? ....!
? Y
```

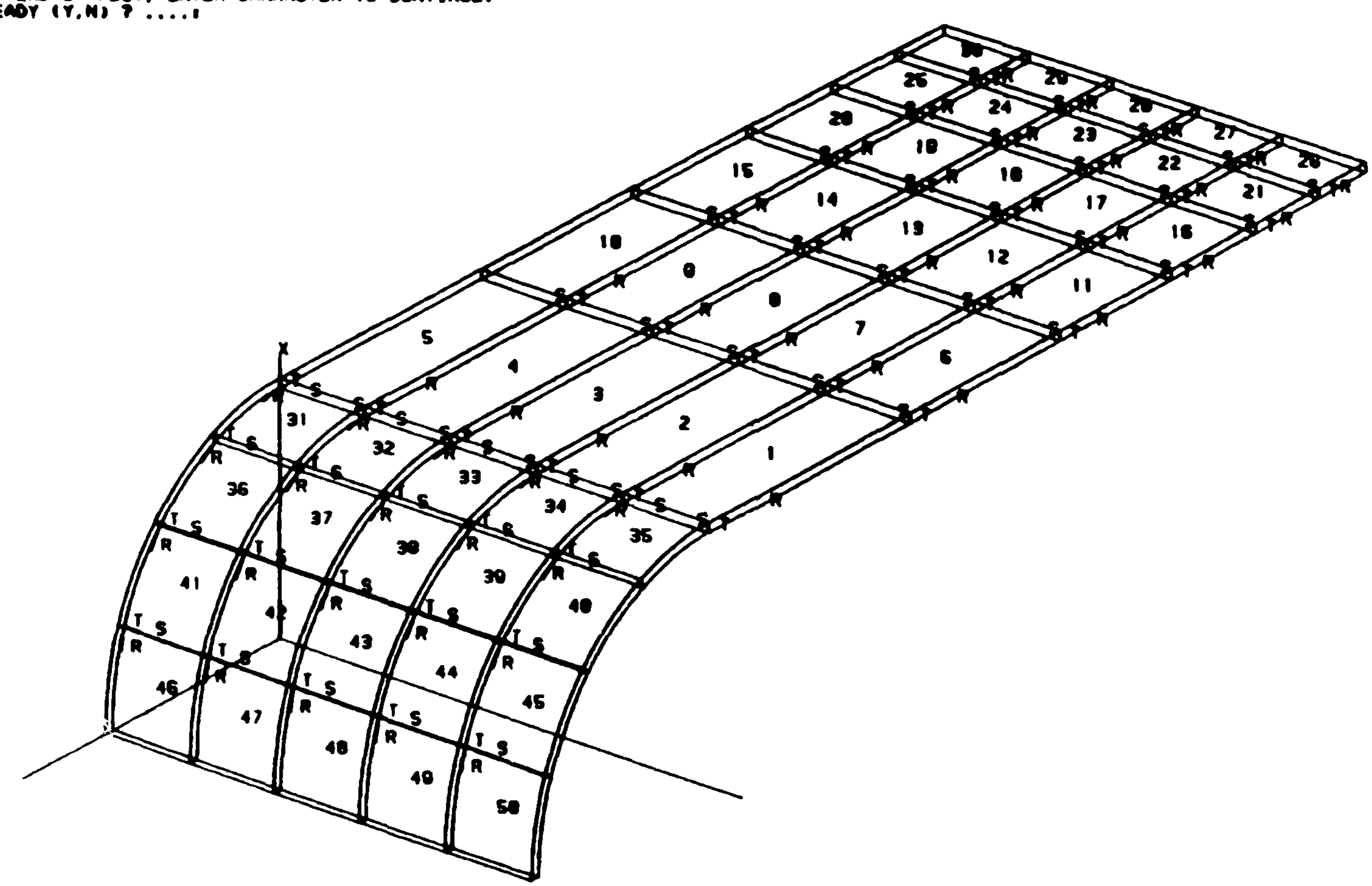


```
MAIN

 ::
? LIS'

 LIS'ING OF DATA FILES AVAILABLE

 #10 --- PLT-ROT
 #12 --- CYL-IGN
 #13 --- PLT-IGN

 ::
? TRANSLATE
 ENTER INPUT FILE #1 LABEL                ::
? PLT-ROT
 ENTER OUTPUT FILE LABEL                  ::
? PLT-RDY
 CREATING NEW DATA FILE
 ENTER TRANSLATION FACTORS(XFAC,YFAC,ZFAC)-----:
? 5.0, 0.0, 10.0

 ::
? MERGE
 ENTER INPUT FILE #1 LABEL                ::
? PLT-RDY
 ENTER INPUT FILE #2 LABEL                ::
? CYL-IGN
 ENTER OUTPUT FILE LABEL                  ::
? BOTH
 CREATING NEW DATA FILE

 TIDY OPTION ONE OR TWO ? (OPT1, OPT2, HELP) ->
? OPT2

       TIDY COMPLETE
    11 DUPLICATE NODES DELETED

 ::
?
PLOT
 ENTER INPUT FILE #1 LABEL                ::
? BOTH
  4
? DRAW

   AT END OF PLOT, ENTER CHARACTER TO CONTINUE.
   READY (Y,N) ? ....:
? Y
```



```
  *
? MAIN

 ::
? LIST

 LISTING OF DATA FILES AVAILABLE

 010 --- PLT-ROT
 011 --- PLT-RDY
 012 --- CYL-16N
 013 --- PLT-16N
 014 --- BOTH

 ::
? DELETE
 ENTER LABEL                  ::
? PLT-ROT

 ::
? DELETE
 ENTER LABEL                  ::
? PLT-RDY

 ::
? DELETE
 ENTER LABEL                  ::
? CYL-16N

 ::
? DELETE
 ENTER LABEL                  ::
? PLT-16N

 ::
? LIST

 LISTING OF DATA FILES AVAILABLE

 014 --- BOTH

 ::
? PLOT
 ENTER INPUT FILE 01 LABEL              ::
? BOTH
  *
? VERT
    WHICH AXIS IS VERTICAL?
    ENTER 1 FOR X, 2 FOR Y, OR 3 FOR Z ..........:
? 1
  *
? EYE
    ENTER THE EYE POSITION,
    XEYE, YEYE, ZEYE ............................:
? 300 500 400
  *
? LABEL
    LABEL THE ELEMENTS?(Y,N) ....................:
? Y
    LABEL THE NODES?(Y,N)  ......................:
? N
```

```
   a
? ORIENT
    PLOT ORIENTATION AXES?(Y,N) .................!
? Y
   a
? DRAW

    AT END OF PLOT, ENTER CHARACTER TO CONTINUE.
    READY (Y,N) ? ....!
? Y
```



```
  *
? MAIN

  ::
? LIST

 LISTING OF DATA FILES AVAILABLE

 #14 --- BOTH

  ::
? RENUMBER
 ENTER INPUT FILE #1 LABEL                    ::
? BOTH
 ENTER OUTPUT FILE LABEL                      ::
? BOTH/R
 CREATING NEW DATA FILE

  NODE POINT REORDERING COMPLETE

  NODAL BANDWIDTH (OLD) =   50
  NODAL BANDWIDTH (NEW) =   50

  ::
? DELETE
 ENTER LABEL                   ::
? BOTH/R

  ::
? BOUNDS
 ENTER INPUT FILE #1 LABEL                    ::
? BOTH
 ENTER OUTPUT FILE LABEL                      ::
? GEOM + BCS
 CREATING NEW DATA FILE

 BOUNDARY CONDITIONS  MAY  BE ENTERED  FOR  RANDOM
   NODES,  RANGES OF NODES,  ALL NODES ON A PLANE,
   OR THE ENTIRE MODEL.

 FOR EACH GROUP OF NODES,  CONSTRAINTS  MAY BE SET
   IN ANY COMBINATION OF THE X, Y, & Z DIRECTIONS.
   (NOTE THAT 2-D MODELS MUST BE FIXED IN Z)

 CONSTRAINT DIRECTION CODE CONSTRUCTED AS FOLLOWS;

   ENTRY 1 - '1' FOR CONSTRAINED IN X DIRECTION
             '0' FOR UNCONSTRAINED IN X DIRECTION
   ENTRY 2 -     SAME FOR Y
   ENTRY 3 -     SAME FOR Z

   EXAMPLE -> '1,0,1' FOR CONSTRAINTS IN X AND Z
                      DIRECTIONS BUT NOT IN Y.

 1 - RANDOM NODES
 2 - RANGE OF NODES
 3 - SPECIFIED PLANE
 4 - ALL NODES
 5 - EXIT

 ENTER NODE SELECTION OPTION (1,2,...5) --------->
? 3
```

```
 ENTER CONSTRAINT DIRECTION CODE (THREE VALUES) ->
? 0 0 1

 PLANE IS DEFINED BY  AX + BY + CZ = D.
   ENTER COEFFICIENTS ( A , B , C , D) ---------->
? 0 0 1 0

              43 NODES FOUND

 1 - RANDOM NODES
 2 - RANGE OF NODES
 3 - SPECIFIED PLANE
 4 - ALL NODES
 5 - EXIT

 ENTER NODE SELECTION OPTION (1,2,...5) --------->
? 3

 ENTER CONSTRAINT DIRECTION CODE (THREE VALUES) ->
? 1 1 0

 PLANE IS DEFINED BY  AX + BY + CZ = D.
   ENTER COEFFICIENTS ( A , B , C , D) ---------->
? 1 0 0 0

              22 NODES FOUND

 1 - RANDOM NODES
 2 - RANGE OF NODES
 3 - SPECIFIED PLANE
 4 - ALL NODES
 5 - EXIT

 ENTER NODE SELECTION OPTION (1,2,...5) --------->
? 3

 ENTER CONSTRAINT DIRECTION CODE (THREE VALUES) ->
? 0 1 0

 PLANE IS DEFINED BY  AX + BY + CZ = D.
   ENTER COEFFICIENTS ( A , B , C , D) ---------->
? 0 1 0 -10.

               0 NODES FOUND

 1 - RANDOM NODES
 2 - RANGE OF NODES
 3 - SPECIFIED PLANE
 4 - ALL NODES
 5 - EXIT

 ENTER NODE SELECTION OPTION (1,2,...5) --------->
? 3
```

```
ENTER CONSTRAINT DIRECTION CODE (THREE VALUES) ->
? 0 1 0

PLANE IS DEFINED BY  AX + BY + CZ = D.
  ENTER COEFFICIENTS ( A , B , C , D) ---------->
? 0 1 0 -20.

              22 NODES FOUND

 1 - RANDOM NODES
 2 - RANGE OF NODES
 3 - SPECIFIED PLANE
 4 - ALL NODES
 5 - EXIT

 ENTER NODE SELECTION OPTION (1,2,...5) --------->
? 5

              07 BOUNDARY CONDITIONS ADDED
               07 BOUNDARY CONDITIONS TOTAL

 ::
? LIST

 LISTING OF DATA FILES AVAILABLE

 #10 --- GEOM + BCS
 #14 --- BOTH

 ::
? DELETE
 ENTER LABEL                ::
? BOTH

 ::
? LOAD
 ENTER INPUT FILE #1 LABEL              ::
? GEOM + BCS
 ENTER OUTPUT FILE LABEL                ::
? ALL BUT PROPS
 CREATING NEW DATA FILE

 LOAD SPECIFICATION OPTIONS
   N - NODAL LOAD ENTRY
   E - ELEMENT LOAD ENTRY
   L - LIST EXISTING LOADS
   H - PRINTS THIS LIST
   S - STOP LOAD ENTRY

 ENTER LOAD SPECIFICATION OPTION    --->
? E
```

```
 ELEMENT SPECIFICATION OPTIONS
   A - ALL ELEMENTS
   S - SINGLE ELEMENT
   R - RANGE OF ELEMENTS
   H - PRINTS THIS LIST
   E - EXIT ELEMENT LOAD SPECIFICATION SECTION

 ENTER ELEMENT SPECIFICATION OPTION --->
? R
 ENTER CASE NUMBER                 --->
? 1
 ENTER SURFACE NUMBER              --->
? 6
 ENTER PRESSURE                    --->
? -1.0
 ENTER BEGINNING ELEMENT,
       ENDING ELEMENT,
       AND INCREMENT               --->
? 21,30,1

 ENTER ELEMENT SPECIFICATION OPTION --->
? E

 LEAVING ELEMENT LOAD SPECIFICATION SECTION

 ENTER LOAD SPECIFICATION OPTION    --->
? S

 LEAVING LOAD SPECIFICATION MODULE

 ::
? LIST

 LISTING OF DATA FILES AVAILABLE

 010 --- GEOM + BCS
 011 --- ALL BUT PROPS

 ::
? DELETE
 ENTER LABEL            ::
? GEOM + BCS

 ::
? TIME
   THE CPU TIME FROM THE START OF THIS SESSION IS    43.182 SEC.

 ::
? PROPS
 ENTER INPUT FILE #1 LABEL           ::
? ALL BUT PROPS
 ENTER OUTPUT FILE LABEL             ::
? FINAL
```

```
CREATING NEW DATA FILE

ELEMENT INPUT OPTIONS
   1)  ALL 2-D ELEMENTS
   2)  ALL 3-D ELEMENTS
   3)  ALL ELEMENTS
   4)  RANDOM ELEMENTS
   5)  RANGE OF ELEMENTS
 ENTER OPTION...:
? 3

 ELEMENT TYPES AVAILABLE...
    0 - DEFAULT (BAR,MEMBRANE,SOLID)
    1 - SHELL OR BEAM
    2 - PLANE STRESS (MEMBRANE)
    3 - PLANE STRAIN
    4 - AXISYMMETRIC
    5 - SHEAR PANEL
    6 - CONTACT ELEMENT
 PRESENT ELEMENT TYPE =    0
 CHANGE (Y,N) ..........:
? N

 MATERIAL TYPE =    0
 CHANGE (Y,N) ..........:
? N

 INTEGRATION OPTIONS AVAILABLE -
    0 - DEFAULT FOR ELEMENT TYPE
    1 - SINGLE POINT
    6 - MIDFACE POINTS
    8 - 2 - PT. GAUSS
   14 - IRON'S RULE
   27 - 3 - PT. GAUSS
 PRESENT INTEGRATION ORDER =    0
 CHANGE (Y,N) ..........:
? Y
 ENTER NEW INTEGRATION ORDER...:
? 14

 ELEMENT PRINTING OPTIONS ----
    0 - PRINTING ON
    1 - PRINTING OFF

 PRESENT PRINTING OPTION =  0
 CHANGE ( Y, N ) ......:
? N

 ::
? STOP

 LISTING OF DATA FILES AVAILABLE

 #10 --- FINAL
 #11 --- ALL BUT PROPS
```

```
PRE-PROCESSOR UTILITIES

  1. AGRID
  2. CORGEN
  3. CREATE
  4. EXPAND
  5. GPLOT
  6. MIDLIN
  7. IJKGEN
  8. NEUTRAL
  9. PREP
 10. REFMT
 11. SPATCH
 12. SURFDIG
 13. TRNSFR
 14. FILE UTILITIES

SELECT BY NUMBER OR TYPE Q TO QUIT ? 14

FILE PROCESSING UTILITIES

  1. ENQUIRE,F
  2. INPUT / OUTPUT FILE NAMES
  3. RENAME
  4. RETURN TO MAIN MENU

SELECT BY NUMBER OR TYPE Q TO QUIT ? 3
ENTER PRESENT FILE NAME? TAPE10
ENTER DESIRED FILE NAME? UNFMT

FILE PROCESSING UTILITIES

  1. ENQUIRE,F
  2. INPUT / OUTPUT FILE NAMES
  3. RENAME
  4. RETURN TO MAIN MENU

SELECT BY NUMBER OR TYPE Q TO QUIT ? 4
```

```
PRE-PROCESSOR UTILITIES

 1. AGRID
 2. CORGEN
 3. CREATE
 4. EXPAND
 5. GPLOT
 6. MIDLIN
 7. IJKGEN
 8. NEUTRAL
 9. PREP
10. REFMT
11. SPATCH
12. SURFDIG
13. TRNSFR
14. FILE UTILITIES

SELECT BY NUMBER OR TYPE 0 TO QUIT ? 10



 ****************************************

 BEGIN REFMT   --   MAGNA INPUT GENERATOR

 ****************************************


 ** SUMMARY OF INITIAL SCAN OF DATA FILE **

 ------------------------------------------

 NUMBER OF NODAL POINTS .......    370
 NUMBER OF ELEMENTS (TOTAL) ...     50
 NUMBER OF CONSTRAINT RECORDS..     87
 NUMBER OF LINEAR CONSTRAINTS..      0
 NUMBER OF NODAL LOADS ........      0
 NUMBER OF ELEMENT LOADS ......      1
 DISTINCT LOAD CASES / GROUPS..      1


    MAGNA         NUMBER OF     ELEMENTS WITH
 ELEMENT TYPE     ELEMENTS     UNSPEC. MATRL

       8              50             50

ENTER A THREE-LINE PROBLEM TITLE   (UP TO 80 CHARACTERS PER LINE)

.........1.........2.........3.........4.........5.........6.........7.........8

?      FOLDED PLATE SEGMENT, DOUBLY SYMMETRIC, PRESSURE BAND AT CENTER
?                   NONLINEAR STATIC ANALYSIS WITH MAGNA
?
```

```
*********************************************
 MAJOR SOLUTION OPTIONS AND PARAMETERS
*********************************************
 ENTER ANALYSIS TYPE      (LINEAR,NONLINEAR) -
? NONLINEAR

 ENTER ANALYSIS SUBTYPE  (STATIC,DYNAMIC)----
? STATIC

 ENTER TIME STEP OPTION  (CONST.,VARIABLE) --
? CONST

 ENTER THE INITIAL SOLUTION TIME STEP -------
? 10.

 ENTER THE NUMBER OF SOLUTION TIME STEPS ----
? 10

 ENTER PRINTING FREQUENCY, IN INCREMENTS ----
? 2

 EQUILIBRIUM ITERATION OPTIONS ARE AS FOLLOWS

    0  =  NO ITERATION
    1  =  MODIFIED NEWTON (CONST. STIFFNESS)
    2  =  FULL NEWTON-RAPHSON ITERATION
    3  =  COMBINED FULL/MODIFIED NEWTON

 ENTER ITERATIVE SOLUTION OPTION (0,1,2,3) --
? 3

 POSTPROCESSOR FILE TO BE WRITTEN (Y/N) -----
? Y

 ENTER THE FREQUENCY (IN INCREMENTS) AT WHICH
  RESULTS ARE TO SAVED ON POSTPROC. FILE ----
? 5

 ARE RESTART FILES TO BE READ     (Y/N) ------
? N

 ARE RESTART FILES TO BE WRITTEN (Y/N) ------
? Y

 ENTER NEW RESTART FILE LABEL (4 CHARS.) ----
? FPLT

 ENTER THE NUMBER OF INCREMENTS BETWEEN
  CHECKPOINTS ON THE NEW RESTART FILE -------
? 2

*********************************************
     END OF OPTIONS SPECIFICATIONS
*********************************************
```

```
INDIVIDUAL ELEMENTS IN THE MODEL CONTAINUNDEFINED PROPERTIES

ELEMENT TYPE       =     8
NUMBER OF ELEMENTS=     50

PLEASE DEFINE A DEFAULT PROPERTY CODE FOR THIS ELEMENT TYPE,
OR ENTER MATERIALS DATA DIRECTLY BELOW.


MATERIAL PROPERTY DEFINITION OPTIONS

 E -- ENTER PROPERTY DATA DIRECTLY
 C -- SPECIFY A LIBRARY PROPERTY CODE
 L -- LIST SELECTED LIBRARY ENTRIES

ENTER OPTION  ( E , C , L ) ------------
? L

LIBRARY MATERIAL DESCRIPTIONS CAN BE LISTED BY MATERIAL TYPE
VALID MATERIAL TYPES ARE AS FOLLOWS ----

ACRYL - ACRYLICS
ALUMI - ALUMINUM ALLOYS
CASTI - CAST IRONS
COPPR - COPPER-BASED ALLOYS
GLASS - GLASSES
MAGNS - MAGNESIUM ALLOYS
NICKL - NICKEL ALLOYS
PLYMR - POLYMERIC MATERIALS
POLYC - POLYCARBONATES
STEEL - CARBON STEELS
STSTL - STAINLESS STEELS
TITNM - TITANIUM

ENTER MATERIAL TYPE (STEEL,STSTL,ETC.)--
? STEEL
```

```
MATL. CODE  ...........DESCRIPTION................

   00100    STEEL - UNS-G 10100   0.1 C    HR
   00101    STEEL - UNS-G 10100   0.1 C    CD
   00110    STEEL - UNS-G 10350   .35 C    HR
   00111    STEEL - UNS-G 10350   .35 C    CD
   00112    STEEL - UNS-G 10350   .35 C    DRAWN 000.F
   00120    STEEL - UNS-G 10500   0.5 C    HR
   00121    STEEL - UNS-G 10500   0.5 C    CD
   00122    STEEL - UNS-G 10500   0.5 C    DRAWN 000.F
   00150    STEEL - UNS-G 41400   0.4C-CR-MO   HR
   00151    STEEL - UNS-G 41400   0.4C-CR-MO   CD

MATERIAL PROPERTY DEFINITION OPTIONS

 E -- ENTER PROPERTY DATA DIRECTLY
 C -- SPECIFY A LIBRARY PROPERTY CODE
 L -- LIST SELECTED LIBRARY ENTRIES

ENTER OPTION  ( E , C , L ) ------------
? C

ENTER LIBRARY PROPERTY CODE --
? 150

MATERIAL PROPERTIES DEFINITION FOR THE MODEL IS COMPLETE.
AT THIS POINT MATERIALS DATA MAY BE EDITED AS NECESSARY.
(NOTE THAT SOME DATA WHICH IS UNIMPORTANT FOR THE CURRENT
ANALYSIS MAY BE DEFINED AS ZERO)

CURRENT PROPERTIES ARE LISTED BELOW ----


CODE  MODULUS   POIS.RATIO  DENSITY   YIELD STR.  THERM.EXP.

  -0  .3000E+08  .2910E+00  .7250E-03  .6300E+05  .7300E-05

L = (L)IST CURRENT PROPERTIES TABLE
C = (C)HANGE AN ENTRY IN THE TABLE
S = (S)TOP EDITING

ENTER OPTION  ( L , C , S ) ------------
? C
(M)ODULUS             (P)OISSONS RATIO
(D)ENSITY             (Y)IELD STRESS
(T)HERMAL EXP. COEFF.

ENTER QUANTITY TO BE CHANGED (M,P,D,Y,T)---
? P

ENTER PROPERTY CODE (AS SHOWN IN TABLE)-
?  0

ENTER POISSONS RATIO ------------------
? 0.20
```

```
L = (L)IST CURRENT PROPERTIES TABLE
C = (C)HANGE AN ENTRY IN THE TABLE
S = (S)TOP EDITING

ENTER OPTION  ( L , C , S ) ------------
? L

 CODE  MODULUS   POIS.RATIO  DENSITY   YIELD STR. THERM.EXP.

  -8  .3000E+08  .2000E+00  .7260E-03  .6300E+05  .7300E-05

L = (L)IST CURRENT PROPERTIES TABLE
C = (C)HANGE AN ENTRY IN THE TABLE
S = (S)TOP EDITING

ENTER OPTION  ( L , C , S ) ------------
? C
(M)ODULUS              (P)OISSONS RATIO
(D)ENSITY              (Y)IELD STRESS
(T)HERMAL EXP. COEFF.

ENTER QUANTITY TO BE CHANGED (M,P,D,Y,T)---
? Y

ENTER PROPERTY CODE (AS SHOWN IN TABLE)-
? -8

ENTER YIELD STRESS ---------------------
? 60000.

L = (L)IST CURRENT PROPERTIES TABLE
C = (C)HANGE AN ENTRY IN THE TABLE
S = (S)TOP EDITING

ENTER OPTION  ( L , C , S ) ------------
? S
```

```
NUMBER OF TIME FUNCTIONS TO BE GENERATED =      1

THESE TIME FUNCTIONS DESCRIBE THE VARIATION OF APPLIED LOADS
DURING THE SOLUTION, BY SCALING THE LOADVALUES GIVEN IN THE
DATA.  LOAD-VS-TIME FUNCTIONS MAY BE GENERATED AUTOMATICALLY
FROM THE OPTIONS BELOW -----

   (1) - STEP FUNCTION (CONSTANT)
   (2) - STEP FUNCTION OVER A SPECIFIED TIME INTERVAL
   (3) - RAMP FUNCTION
   (4) - TRIANGULAR PULSE
   (5) - USER DEFINED


 LIST ELEMENT PRESSURE DATA ( Y , N ) ---
? Y

  E L E M E N T   P R E S S U R E   D A T A

    FIRST LAST INCR SURF CASE  PRESSURE
      21   30    1    6    1 -.10000E+01

              ***** DATA CURVE OPTIONS *****

 I                          I
 I     ( OPTION 1 )         I       ( OPTION 2 )
 I                          I
 XXXXXXXXXXXXXXXXXXXXXXXX   XXXXXXXXXXXX
 X                          X          X
 X                          X          X
 X                          X          X
 X-----------------------   X----------XXXXXXXXXXXXXXXX
 I                   X      I             X
 I                X         I          X     X
 I             X            I        X         X
 I          X               I      X             X
 I       X                  I    X                 X
 I    X                     I  X                     X
 I  X     ( OPTION 3 )      I X     ( OPTION 4 )       X
 X-----------------------   X----------------------------X
( OPTION 5  =  USER - DEFINED DATA CURVE )

 ENTER TIME FUNCTION TYPE FOR CURVE NO.   1 -
? 3

 PEAK LOADING VALUES ARE ASSUMED TO BE THOSE SPECIFIED IN THE
 ORIGINAL DATA FILE.  DO YOU WISH TO SCALE THESE VALUES (Y/N)
? N


 L = (L)IVE ELEMENT SURFACE PRESSURES
 D = (D)EAD LOADING

 ENTER SURFACE PRESSURE TYPE ( L , D ) --
? L
```

```
****************************************
*                                      *
*      DATA GENERATION COMPLETE        *
*                                      *
****************************************

PRE-PROCESSOR UTILITIES

 1. AGRID
 2. CORGEN
 3. CREATE
 4. EXPAND
 5. GPLOT
 6. HIDLIN
 7. IJKGEN
 8. NEUTRAL
 9. PREP
10. REFMT
11. SPATCH
12. SURFDIG
13. TRNSFR
14. FILE UTILITIES

SELECT BY NUMBER OR TYPE Q TO QUIT ? Q
/REWIND,FDATA
REWIND,FDATA.
/DEFINE,MYFILE
/COPYBF,FDATA,MYFILE
 COPY COMPLETE.
/CHANGE,FOLDPLT=MYFILE
/
```

REVISED POSTPROCESSOR ACCESS PROCEDURES UNDER NOS
(CDC Program Versions)

INTERACTIVE ACCESS THROUGH CCL PROCEDURE "UTILS"

```
GET,UTILS/UN-DB20139
/GET,META
/UTILS

 PROBLEMS WITH MISSING PROGRAM FILE (FILE NOT FOUND)
 SHOULD BE REFFERED TO  TSGT. S. ZASTROW, 855-6190.


 MAGNA PRE/POST PROCESSOR UTILITIES

  1. PRE-PROCESSORS
  2. POST-PROCESSORS

 SELECT BY NUMBER OR TYPE Q TO QUIT ? 2

 POST-PROCESSORS

  1. CPLOT (DISSPLA OPTION)
  2. CPLOT (APOST VERSION )
  3. CPLOT
  4. HIDLIN
  5. URTFIL / XYPLOT
  6. UTFILA (APOST DATA CONV.)
  7. XYPLOT
  8. FILE UTILITIES

 SELECT BY NUMBER OR TYPE Q TO QUIT ? 1

   ##########################
   BEGIN CPLOT\DISSPLA OPTION
   ##########################

 ENTER DESIRED MODEL
? 4014
 ENTER LINESPEED (CHARACTERS PER SECOND)
? 120
 ENTER RESOLUTION MODE (0-LOW)
? 1
 ENTER OPTION(0-DEF, 31-8 PENS ON 4382, 36-PAPER FEED ON 4663)
? 0
  ENTER POST-PROCESSOR DIRECTIVES
?
```

BATCH INPUT STREAM FOR CPLOT WITH DISSPLA OUTPUT OPTION

```
/JOB
UJN,STCSB.
/USER
CHARGE,*.
COMMENT.  #############
COMMENT.  ## NO DECK ##
COMMENT.  #############
GET,CPLTDIS/UN-DB20139.
GET,TAPE8= POST.
SETTL,30.
CPLTDIS.
SAVE,META.
ENQUIRE,F.
/EOR
 0
 0
MOVIE
Y
Y
 1 4 1
DEFORM
 2
Y
 1.0
VERTICAL
 2
EYEP
Y
 0 0 10000
CORE
 1
 2.6E+05 2.93E+06 5.0E+05
N
 0
ENTE
Y
CUBE
Y
0.00603 5.0 0.0 2.5 0.0 6.25
Y
MDRAW
Y
 1 4 1
EXIT
/EOR
/EOF
END OF FILE
??
```


U.S.Government Printing Office 1986 646-067/40829